Physics

COO-1469-0111
COO-1018-1164

1/4

QUARTERLY TECHNICAL PROGRESS REPORT

July, August, September 1968

DEPARTMENT OF COMPUTER SCIENCE
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN · URBANA, ILLINOIS

QUARTERLY TECHNICAL PROGRESS REPORT

July, August, September 1968

Department of Computer Science
University of Illinois
Urbana, Illinois 61801

TABLE OF CONTENTS

TABLE OF CONTENTS (CONT'D)

TABLE OF CONTENTS (CONT'D)

TABLE OF CONTENTS (CONT'D)

TABLE OF CONTENTS (CONT'D)

1. CIRCUIT RESEARCH

(Supported in part by the Office of Naval Research under Contract NR 048-102/6-15-67 Code 437, W. J. Poppelbaum, Principal Investigator).

Summary

John Esch reports on the status of the Stochastic Array Computer and describes a circuit for generating clocked random pulse sequences of given average frequency.

In the new project of Bundle Processing David Ring discusses problems of implementing arithmetic operations.

M. Faiman

1.1 Random Sequence Coding (Project No. 03)

1.1.1 CRPS Generator

During the summer the circuit for generating a CRPS was finalized and is shown in Figure 1. The chief elements in the circuit are a noise source, a threshold detector, a sampling unit and an error integrator.

Pulses which are random in both amplitude and time are generated by the (transistor) noise source and fed to one input of the threshold detector. The other (threshold) input is provided by the output of the error integrator. Only if the noise voltage exceeds threshold is the output of the threshold detector a "1," and this is sampled by a clocked flip-flop whose output constitutes the CRPS. The latter is fed back to the error integrator for comparison with the analog input signal.

A minor drawback of the system is the non-linearity of the error integrator, which results in the CRPS value not being proportional to the analog input voltage. This could be overcome by increasing the gain of the error integrator and adding more circuitry. However, linearity is not a requirement in the initial application: each CRPS generator's output is set by monitoring it with a frequency counter and adjusting the analog input to give the desired output. Once set, it remains fixed by the action of the error integrator.

1.1.2 Layout of Stochastic Array Computer

For experimental test extensive control and monitoring features are provided by an "information panel" in the Stochastic Array Computer. All CRPS generators and computing elements are mounted on printed circuit cards, and the location of each such element is indicated by a panel light. The panel also has switches for controlling the arithmetic operations to be performed. Each card is provided with a set of test "fingers" to which a monitoring connector may be attached. The monitoring system comprises decoding and buffering circuitry to drive the counter and a subset of the panel lights to indicate those elements being monitored.

John Esch.

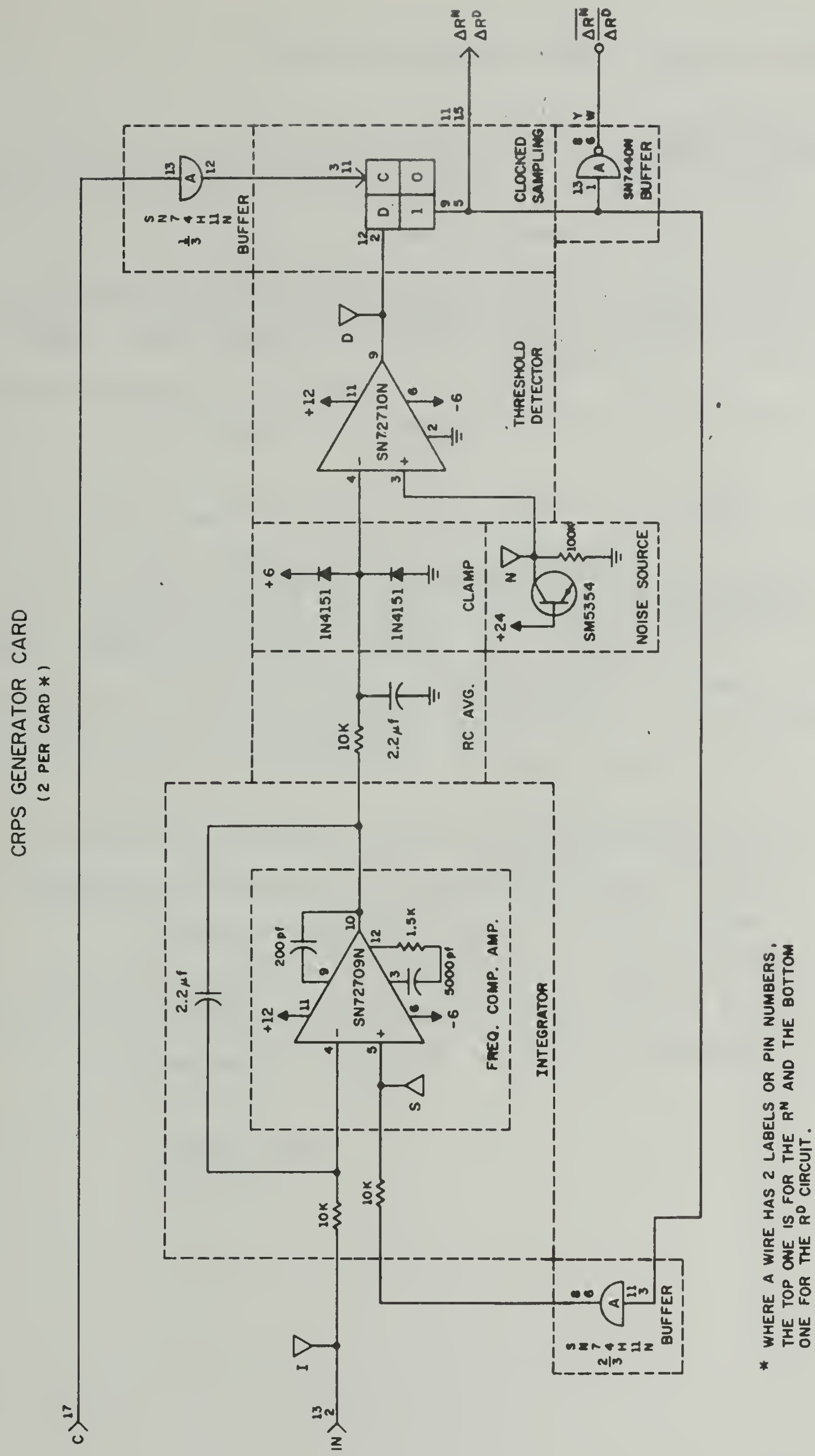


Figure 1. CRPS Generator

1.2 Bundle Processing (Project No. 21)

1.2.1 Current Status

Bundle processing is an application of stochastic methods to the spatial, rather than the temporal domain. If n wires, randomly chosen from a larger bundle of N wires, carry ones, the remainder being zero, then n/N (or some linear function of this quantity) can be used as a representation of some number x. For example,

$$x = \frac{2n}{N} - 1$$

yields a representation of N numbers in the range (-1, 1).

If two such bundles are combined, the representation of the sum is

$$y = \frac{2(n_1 + n_2)}{2N} - 1 = \frac{1}{2}(x_1 + x_2)$$

which performs arithmetic addition nicely, aside from the factor of one-half.

Currently under investigation is a device for doubling the number of ones in a bundle so as to remove this unwanted factor.

Another problem being looked into is that of division. A mapping that makes use of two bundles per number, one each for numerator and denominator, is being examined and will be reported on subsequently.

David Ring.

2. HARDWARE SYSTEMS RESEARCH

(This work is supported in part by Contract No. AT(11-1)-1469 of the Atomic Energy Commission and in part by the University of Illinois.)

Summary

Doug Sand reports on the overhaul of parts of OLFT's electronics, to which the summer period was devoted. Most of the troubles were in the smaller equipment cabinet and have now been fixed.

VISTA is nearing completion--David Rollenhagen contributes a status report.

In the Functional Encoding project, most of the summer's work, described by Ed Carr, has centered on the A/D converter.

Larry Ryan, Orin Marvel and Yiu Wo describe the requirements for, and preliminary design of an output display for Transformatrix.

Finally, Shiv Verma gives an initial account of a new, three-dimensional display project, Stereomatrix, which will be tackled jointly by both hardware and software people from the 1469 contract.

M. Faiman

2.1 OLFT (Project No. 12)

2.1.1 System Modifications

Several minor system failures have occurred. In particular, system components mounted in the samll rack (located on the optical bench) have performed erratically or failed altogether. This equipment rack contains the yoke driver, referenced to ground potential, and the video isolation amplifier, referenced to cathode potential (-25KV). The high voltage chassis is inadequately isolated, resulting in electrostatic discharges and strong high-frequency radiation. Many of the failures have been traced to this lack of isolation, as well as several minor defects, such as imporper wiring. Accordingly, the components in the small rack are being rearranged to improve the high-voltage isolation and to increase component accessibility. Additional modifications are in progress, including the provision of a digital bar-pattern generator for resolution tests and a precise target-current monitor.

2.1.2 Light Valve Resolution

A first-order analysis of the resolution characteristics of the light valve has been developed* and will be discussed in detail in a subsequent report. In essence, this analysis predicts that the resolution N in line-pairs is very roughly, $N \sim t/T$, where t is the length (normal to the lines of the bar-pattern written of the crystal face, and T is the crystal thickness. Our crystals are typically 1" in diameter and about 10 mils in thickness, so we might expect a resolution of about 100 line-pairs. This prediction agrees with previous observations, and a series of resolution measurements will be undertaken to test these results.

The crystals we are using are the best that are commercially available. The vacuum chamber limits the crystal diameter to about 1",

---

*D.S. Sand, M.S. Thesis (to be published)

and we have not been able to obtain a crystal of this diameter polished to less than about 10 mils in thickness. This limitation in thickness may be attributed to difficulties in polishing, since KDP is a very "soft" crystal.

We are currently attempting, by several parallel efforts, to improve the resolution of the light valve to at least 300 line-pairs. In particular, we are searching for a thinner crystal, either of KDP (using better polishing techniques), or of some other electro-optic material which is less soft and more easily polished to a smaller thickness. Further, since the light valve acts as a low-pass spatial filter, we are investigating the possibility of pre- or post-emphasis of the high-frequency content of the resulting picture. Pre-emphasis could be obtained by suitably processing the video signal before it reaches the electron gun, and post-emphasis by an appropriate spatial filter located in the optical Fourier transform plane, or possibly by electronic processing of the video signal of a vidicon recording the spatial Fourier transform. The problem of resolution and methods for its improvement will be considered in more detail in a subsequent report.

David Casasent
Douglas Sand

2.2 VISTA (Project No. 14)

2.2.1 System Status

The cabinet which encloses the VISTA System has been completely wired with the exception of the control panel. Four of the six subsystems have been completed and are functioning. These are the A/D Converter, the Horizontal Ramp Generator, the Frame Control Circuitry and the Vertical Ramp Generator. The Scan Program Control subsystem is being completed and, when ready, the design of the variable-scan television camera will begin.

David Rollenhagen

2.3 Functional Encoding (Project No. 15)

2.3.1 A/D Converter

As mentioned in the last QTPR, a means of subtracting the analog equivalent of the first 5 bits of the A/D converter from the analog voltage stored in the sample and hold circuit was needed. This circuitry was designed and built this past quarter. The circuit, shown in Figure 1, works as follows. Input A is an analog current equivalent to the first five bits. Input B is the analog voltage stored in the sample and hold circuit. The 50Ω potentiometer on input B serves to calibrate the current into the summing node of the amplifier. The amplifier resistive feedback network serves to set the gain of the amplifier: it is adjusted until the maximum output, due to the difference in currents at the input, is .250 V. The purpose of the 1N914B diodes in the feedback network is to provide a limit to the output voltage during the period when input A has not arrived and input B is already there; this prevents the Nexus amplifier from saturating. The 2N 2905A transistor serves as an emitter follower and provides a lower output impedance to the feedback and second stage amplifier circuits. The second stage amplifier inverts and correctly scales the difference voltage generated by the first stage.

In addition to the completion of the A/D Subtracter-Inverter circuit, the entire A/D Converter has been built on printed circuit boards and is being tested, prior to its installation in the Functional Encoding System.

Ed Carr.

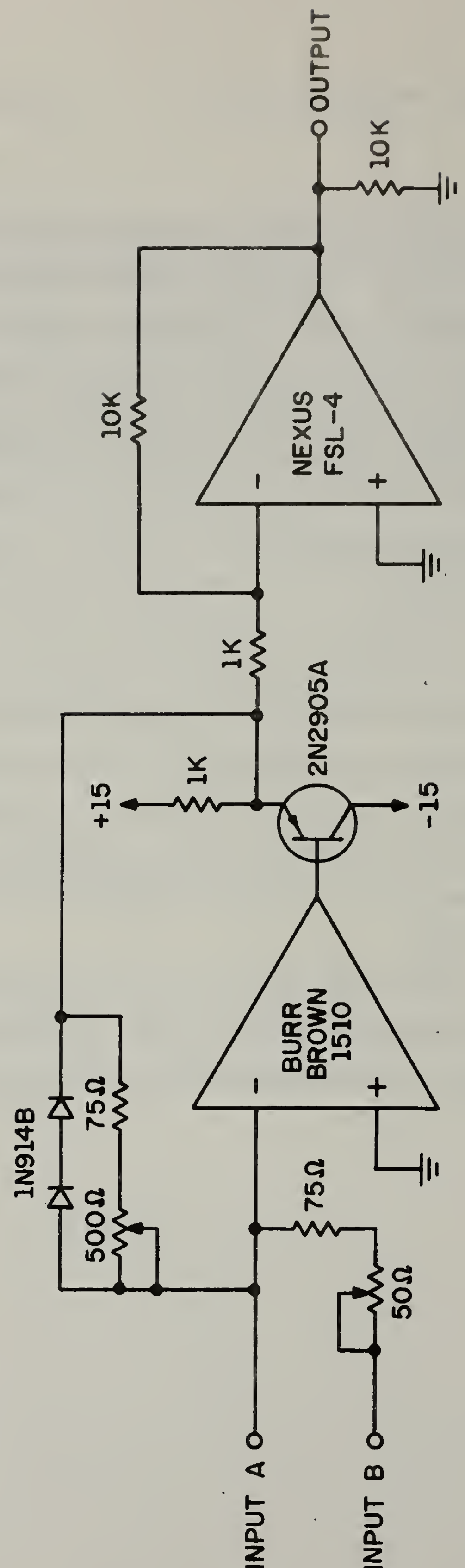


Figure 1. A/D Subtracter-Inverter

2.4 Transformatrix (Project 17)

2.4.1 Output Display

A considerable amount of effort has been spent this summer on selecting a suitable output display system for Transformatrix. As mentioned in the previous report, Transformatrix transforms the input picture, in the form of a square matrix of 32 x 32 sample points, into an output picture of the same format, at the rate of 30 pictures per second. Therefore, the general requirements on the output display part of Transformatrix are as follows:

(1) The display device must be able to produce a matrix of 32 x 32 elements at the required rate.

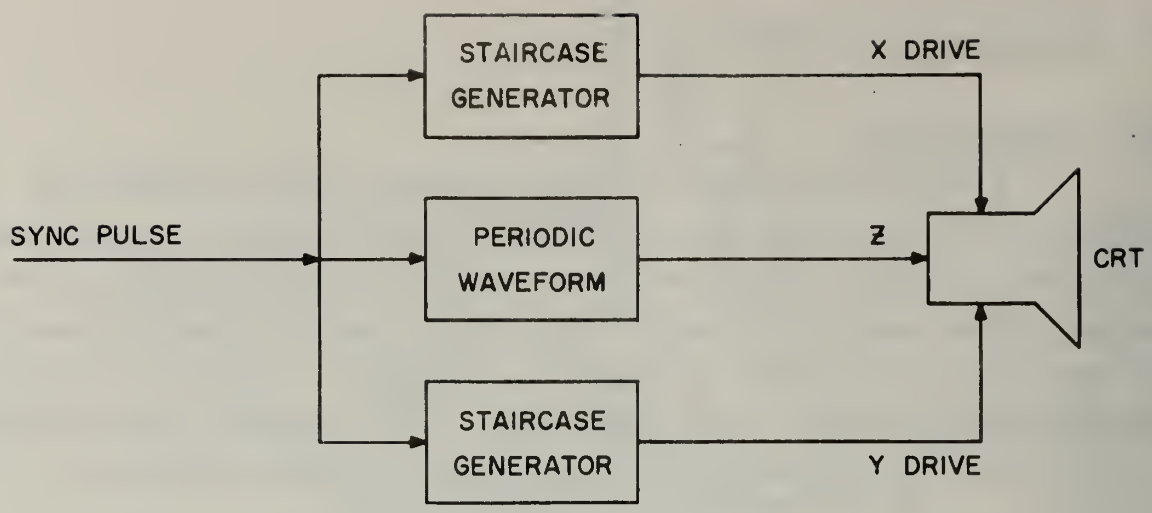
(2) The physical dimensions of the output picture, as well as the size and shape of the elements, must be suitable for viewing at a distance of a few feet.

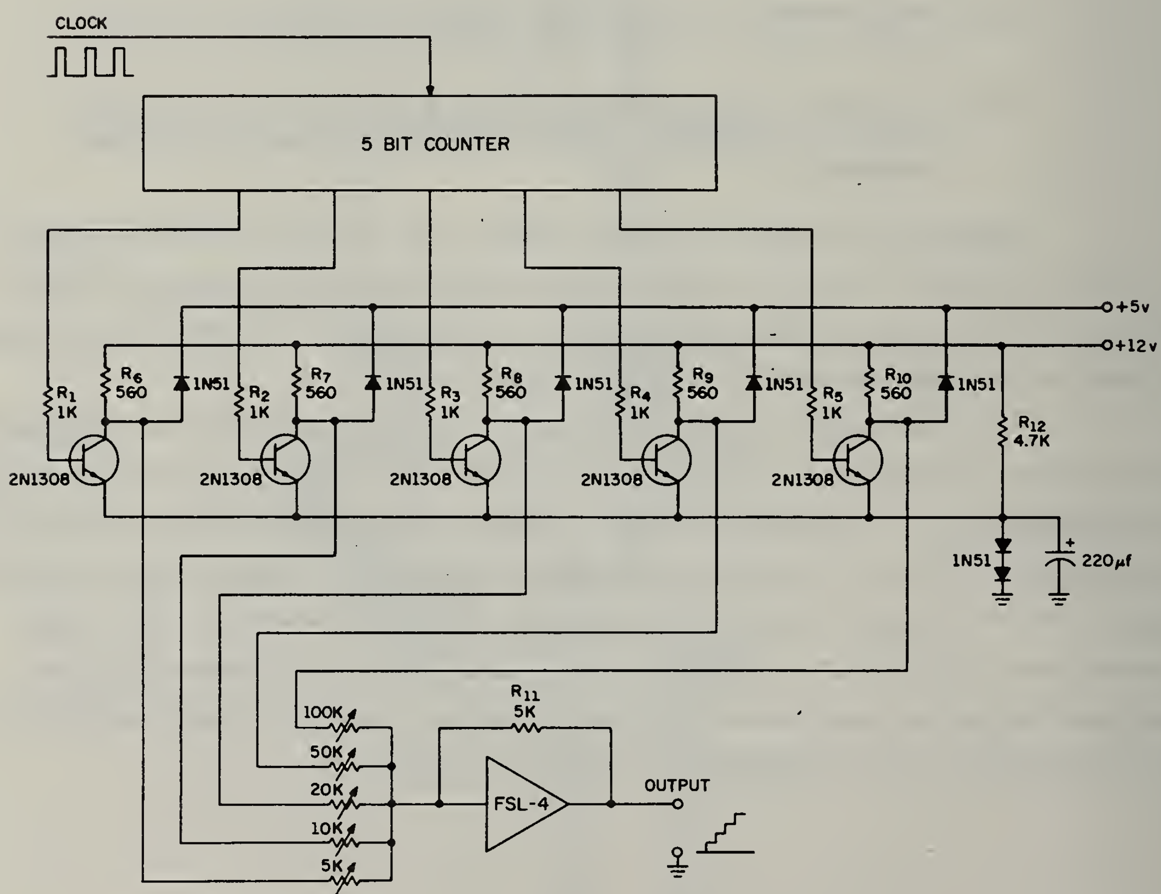
(3) Each element of the matrix must be sharp; its size and shape must be uniform over the entire matrix.

(4) The grey level resolving power must be not less than 8 levels as required by the overall performance of Transformatrix.

Amoung the various display devices, the cathode ray tube appears to be most suitable. Work is proceeding on gathering data, through experiments as well as correspondance with some manufacturers, in order to be able to make an optimum choice on a specific type of CRT.

An experiment designed to study the grey levels and the appearance of the output format as well as dynamic behavior of a defocused electron beam is now under way. A simulated output picture is produced on the screen of an oscilloscope. This is obtained by driving the x and y axes of the oscilloscope with two 32-step staircase waveforms. The grey levels of the output elements are produced by modulating the z axis with a synchronous periodic waveform having the same period as that of either the x or y axis waveforms. A block diagram of this apparatus is shown in Figure 1.





(upper) Figure 1. Block Diagram of the experiment

(lower) Figure 2. Circuit Diagram of the Staircase Generator

The actual circuit used for the staircase generator is shown in Figure 2. A 5-bit counter is driven by a clock of 30KHZ frequency. The outputs of this counter are standardized to 5 volts through the transistors 2N1308 and the diodes 1N51. Then they are weighted according to their bit number and summed by the adder using the variable resistors, R_{11} and the operational amplifier FSL-4. The resistor R_{12} forward biases the diodes to raise the emitter voltage of all 2N1308's to about 1.0 volt.

This will ensure that a 2N1308 is cut off when the corresponding output of the counter is at '0'.

Larry Ryan
Orin Marvel
Yiu Wo

2.5 Stereomatrix (Project No. 30)

2.5.1 Introduction

Stereomatrix will be a system for the three-dimensional display of computer generated curves. It may be regarded conceptually as consisting of an input section, a processor and a display.

The input section receives point coordinates (x, y, z), and possibly control information, in digital form from a computer via a data disc. The processor is capable of performing a translation/rotation operation on each point, and will then generate a stereoscopic pair, (x_L, y_L) and (x_R, y_R), suitable for viewing. Feedback information for the computer will also be provided by the processor. The stereoscopic pair is subsequently displayed in an appropriate manner.

The project is to be a cooperative venture between the hardware and software groups of the 1469 contract, so that many details will not be resolved until further discussion. It is envisaged that the processor will consist of fast hybrid circuitry, but the exact location of the digital-hybrid interface is not yet determined. Other items to be finalized are data rates, the physical nature of the display, the type and amount of buffer storage, and the extent of the interaction between viewer and computer.

2.5.2 Stereoscopic Pair

The formation of a stereoscopic pair is shown in Figure 1. For simplicity, the image plane is taken as Z = 0; P_L and P_R constitute the stereoscopic pair, formed by the intersections of PL and PR with the image plane. The geometry is elementary:

$$\frac{x_L}{x'} = \frac{x_R}{x'} = \frac{y_L + a}{y' + a} = \frac{y_R - a}{y' - a} = \frac{1}{1 + z'},$$

so that

$$x_L = x_R = \frac{1x'}{1+z'}; \quad y_L = \frac{1y' - az'}{1+z'}; \quad y_R = \frac{1y' + az'}{1+z'}$$

These transformations can be implemented by fast hybrid circuitry in a fairly straightforward fashion.

Shiv Verma

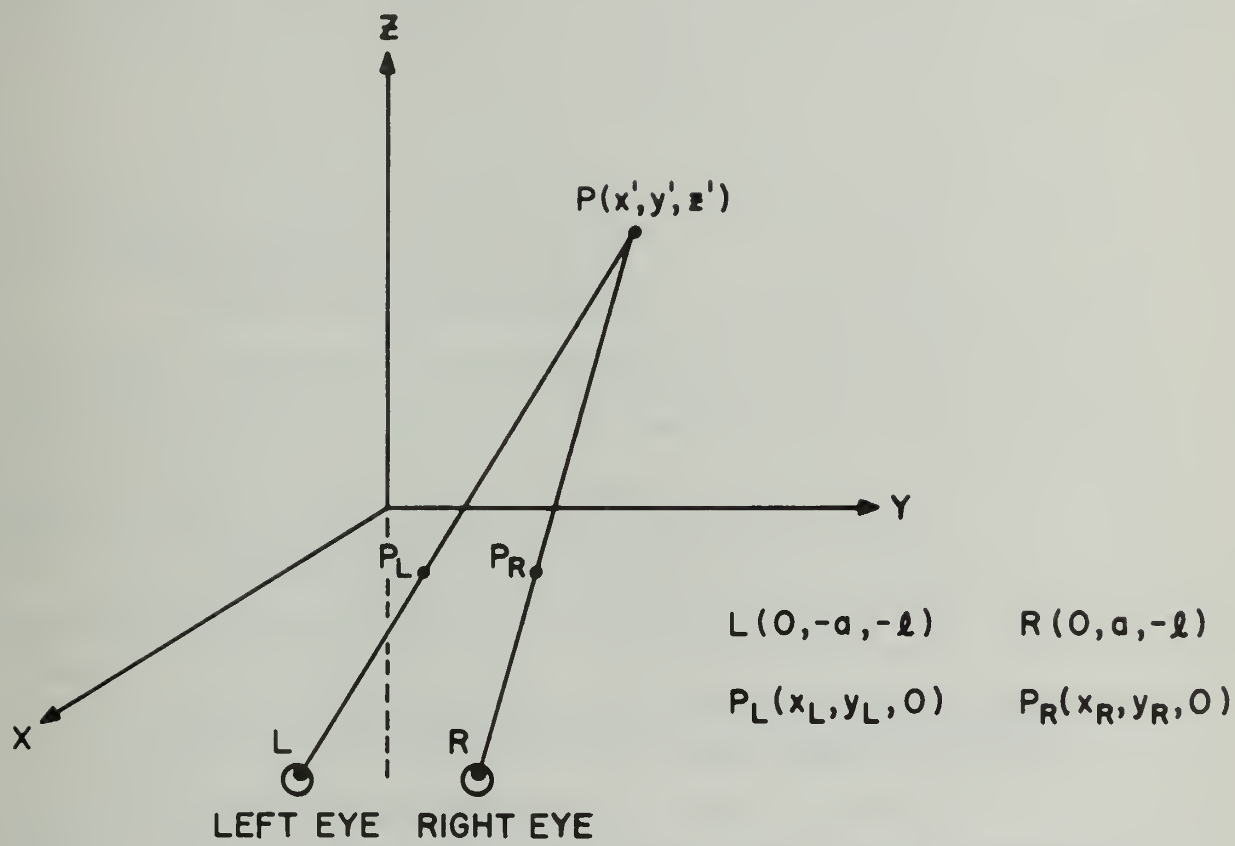


Figure 1. Formation of Stereoscopic Pair

3.1 Ordinary Differential Equations

The program to integrate ordinary differential equations has been altered. A new key word has been added for the purpose of allowing the user to specify the number of times per page he would like the results of the integration to be printed. This key word may be used whenever the user specifies a page heading and a format for printing results. This gives the user complete control over the format of the output, if he so desires. In order to use this option the user punches the word LINES in a data card, followed by the number which is the desired number of printings per page of output. The number may be written in any standard form; the Supervisor program interprets all numbers as floating point quantities.

Corrections made in the program are the following:

1. There had been a restriction that whenever a differential equation or print expression began with two consecutive letters, e.g., sin(x), the expression had to be enclosed in parentheses. This restriction has been removed.

2. Previously a differential equation involving a unary minus such as y' =-sin(x) caused inaccurate solutions. This has also been corrected.

3. Print expressions, other than the solutions to the equations, caused the solutions to be incorrect even though the values of the print expressions were calculated correctly. This solution had to do with the presence of constants in the print expressions and has been eliminated so that both the solutions and print expressions are calculated correctly.

3.1.1 Graphic System for Studying High Order Methods for Stiff Differential Equations

An interactive graphic display system has been designed for use in connection with the study of new methods, especially of degree greater than 6, for numerically solving stiff differential equations. Considering the usual form for multistep methods

$$\alpha_0 y_{n+1} = -\left[\sum_{i=1}^{k} \alpha_i y_{n+1-i} + h \sum_{i=0}^{k} \beta_i y'_{n+1-i}\right]$$

The required input to the system will be k and the coefficients β_0 through β_k. Specifying the β's enables the determination of $\sigma(\xi) = \sum_{i=0}^{k} \beta_i \xi^{k-i}$ and the unique method of degree k for this $\sigma(\xi)$, i.e. α_0 through α_k.

Given this, we obtain $\rho(\xi) = \sum_{i=0}^{k} \alpha_i \xi^{k-i}$. The stability of the method can be studied by examining the locus of $\frac{\rho(\xi)}{\sigma(\xi)}$ in the $h\lambda$ plane for $\xi = e^{i\Theta}$ where $\Theta = [0,2\pi]$. This locus will be displayed on the 338 CRT display. All calculations will be done on the PDP-8, and the DEC floating point package will be used for all floating point arithmetic. Basic interactive capabilities will provide for changing one or more of the $\beta(i)$'s or k at any time. A complete system description will be given in a later report.

C. Dill
C. Ellis
K. Ratliff

3.2 DEC 338 Software

Work was on the development of a more general display system for the DEC 338, which will provide service to rather diverse users. In this system the user is allowed to create his own pictures or symbols as on the light-button and on the light-button the user can create in the computer a model of his design problem.

A data structure for this proposed general purpose display system package has been derived, and the implementation of this system has been started. The first phase of this implementation is to provide the facilities to help the user create or define his own light-button so that a model of his design problem can be built by using those light-buttons with the operations provided by the system.

Any symbol or figure which can be drawn on the display screen with points and lines can be defined as a light-button. The system will enable a user to create a light-button using the light pen and the function push-buttons. Once the light-button is drawn, the user is required to specify points on the light-button as connecting points if this light-button is to be connected with other light-buttons when the user creates his model

of design problem in the computer. In this system the connection is classified into two types: one is the free (connection) type and the other is the restricted (connection) type. In the free type the two light-buttons are connected at their closest connecting points. In the restricted type the two light-buttons are connected at the two connecting points already specified by the user. If one of the light-buttons to be connected is of free type then the shortest distance criterion will be used in selecting the connecting point of the free type connecting light-button. A user is therefore required to declare the type of more than one connecting point on this light-button. Text is allowed to be inserted into the light-button as a permanent part of it.

A completely defined light-button can be rotated to $\pi i/2$ degree ($i = 1,2,3$), and each of these rotations will be considered by the system as a different light-button, but a table of equivalent light-buttons will preserve their equivalence.

A detailed description of techniques for creating light buttons will be provided after the coding, debugging and evaluation of this piece of system are completed.

T. Lo

3.3 Flowchart Programming Language II (FPL/II)

FPL/II is a language implemented in the experimental version of CAPS for use on the S/360. FPL/II provides facilities for compiling and executing a program in a flowchart media. FPL/II receives a flowchart converted to digital data from the PDP-8, creates a table for the digital information, reconstructs the flowchart in a list form, and executes it. The intermediate table of digital information can be stored on an auxillary storage device so the flowchart can be executed at a later date upon request without reconstruction. The description of this table and the flowchart list format has been described in an earlier report.

FPL/II is a list oriented language with basic building blocks called elements. Element length is always a multiple of 4 bytes, i.e. a word. The format for all elements requires that two fields always have the same usage, and that the remainder of the element be determined by the contents of these two fields and the context in which the element appears. The two constant fields are 1) the type--an 8-bit specification that implies the format of the rest of the element, and 2) a horizontal linkage address--a 24-bit address that links this element to the next one in the list; the field is zero if this is the last element in the list.

The following types have been implemented:

1. Pointer: The pointer element is two words long, and the second word is a linkage address to another element not necessarily in the same horizontal list. The context in which the pointer is encountered determines how the linkage is used.

2. Marker: A marker element, two words long, is used to delimit the end of a list associated with the execution of a particular flowchart. Further comments on its use will follow.

3. Null: The null element, two words long, is a place-holder on which flowchart execution lists are built, and when encountered, signals that all flowcharts invoked have been executed.

4. Character String: A character string element indicates a variable length element which contains a string of characters which may be either an identifier name, a flowchart name, a primitive name, or a string of text from a flowchart symbol. The second word contains the length of the text which begins in the third word and is given in bytes. The length must be a multiple of 4 bytes; the text is padded with blanks if necessary.

5. Fix: This element is two words and the second word is a fixed point number.

6. Float: This element is three words long, the second and third of which contain a double precision floating point number.

7. <u>Empty</u>: This element can occur only in the free list (described in an earlier report). The length of the element beginning with the third element is in the second word.

8. <u>Released</u>: This type is superimposed over the other types (which imply the element length) and indicates that the element is no longer active. When the garbage collection routine is next run, the released elements can be added to the free list.

9. <u>Primitive</u>: This is a two word pointer element, the second word of which links directly to the beginning of a primitive routine in memory.

10. <u>Flchdisk</u>: This element is two words, the second of which is a disk address of a flowchart table.

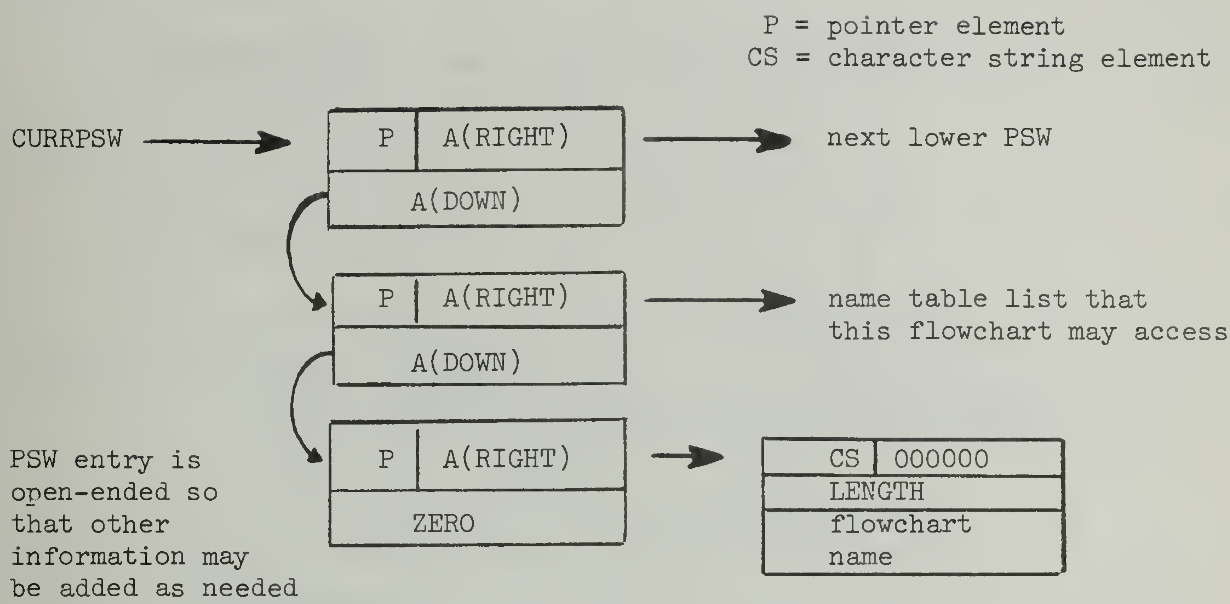
11. <u>Flchlist</u>: This element is also two words but the second is the address of the first executable element in a flowchart list.

There are five major lists controlled by FPL/II. All lists have a one-way linkage, and access to these lists is through the constant area common to all flowcharts, primitives, and system routines.

1. <u>The Flowchart List</u>: This format has been discussed in earlier reports.

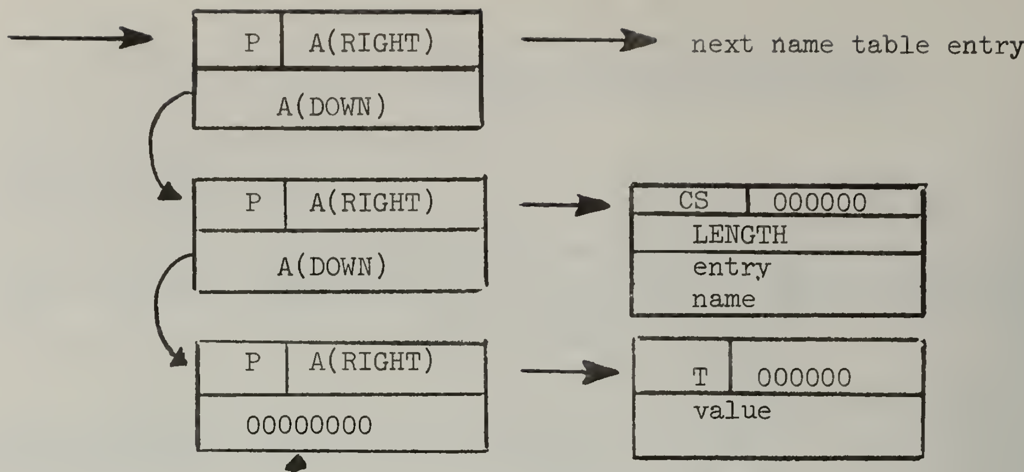
2. <u>The PSW List</u>: The system control program has a permanent entry in this list, and each flowchart which is called for execution has an entry placed in the list. The list behaves as a stack. The most recent entry is the current active PSW, and it is linked to lower levels of PSW's which are passive. When the execution of a flowchart is completed, the PSW for that flowchart is deleted. The term PSW was borrowed from the S/360 terminology and implies that pertinent status information is contained therein. Access to the PSW list is made through the common constant area in the field called CURRPSW.

An active PSW entry format is:



3. <u>Name Table List</u>: The name table list contains identifiers and their values, primitives and their memory locations, flowcharts and their disk addresses or flowchart list addresses. The name table for a flowchart is linked to the PSW for that flowchart. All flowchart name tables are linked to the system name table. If a flowchart is invoked by another flowchart their name tables are linked together, and thus to the system name table, provided the scope of the variables of the calling flowchart includes that of the called flowchart. Entries are placed in the name table during execution of a flowchart either by use of the SET primitive or by designation of certain variables to be input parameters from another flowchart. Entries are deleted by the RELEASE primitive or by completion of the execution of the flowchart. Variable names are referenced on a first-in-first-out basis, so a reference to a variable name always is a reference to the most recent entry.

The format of a name table entry is:



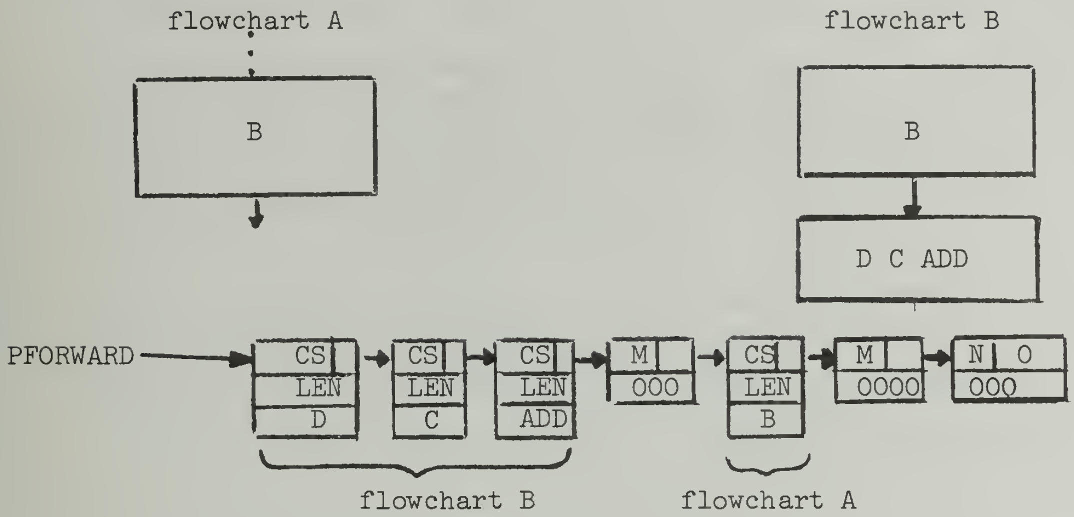
The name table entry is open-ended so that other information may be added as needed

T = the type associated with the entry and designates what the value is

4. Pointer-Forward List: The PF list is initialized by the system control program with a single element, the Null element, discussed earlier. Before executing a called flowchart a Marker element is added to the PF list to delimit that portion of this list associated with the new flowchart. When the Executor is ready to execute the next symbol in the flowchart list, it calls upon the Recognizer to break the text in that symbol into components, put the components into a list in the order of occurrence, and link that list to the PF list. The address of the beginning of the list is then stored in the constant area field called PFORWARD. As each element of the list is analyzed by the Executor and acted upon, that element is then released, and the address in PFORWARD is updated to the next symbol. When the Marker element is encountered execution of that flowchart list symbol is completed and the Executor moves to the next symbol in the flowchart list. When the flowchart is completed the Marker symbol is deleted from the list and the Executor analyzes the next element in the list which may belong to a higher level flowchart or may be the Null element which signifies that all jobs are completed.

As an example, suppose flowchart A calls flowchart B (without parameters for simplicity). The PF list looks like the following picture

after the recognizer has broken up the elements in the first executable symbol of flowchart B.

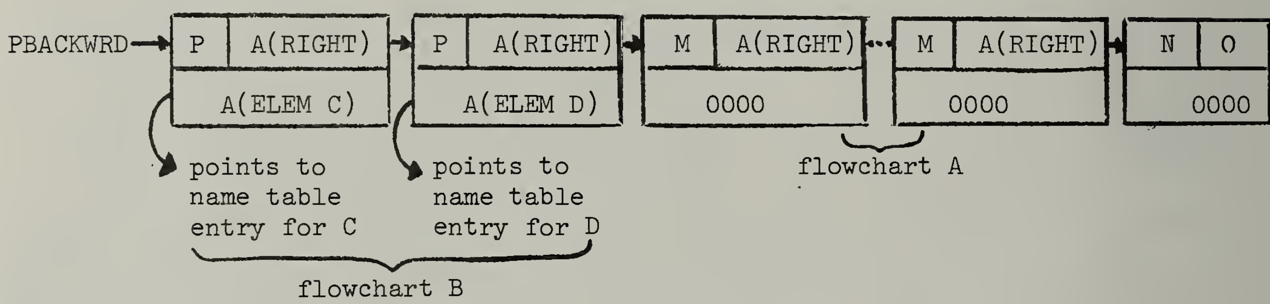


M = marker element
N = null element
LEN = element length

5. <u>Pointer-Backward List</u>: The PB list resembles the PF list physically with its initial Null element and the addition of a Marker element for the execution of each new flowchart. The location of the PB list is available in the constant area field called PBACKWRD. As the executor is analyzing the elements in the PF list it will find one of two types, a CS type, as illustrated above, or a numerical type, FIX or FLOAT. The numerical types occur when text such as 'A 3 ADD' occurs in a flowchart symbol. If the element is a numerical type the element is copied into the PB list exactly as it is, and is released from the PF list. If the type is CS the executor does a name table search beginning from the entry designated by the PSW for the current flowchart. If no match is found the CS element is copied into the PB list and is deleted from the PF list. If a match is found, further investigation of the type is necessary. If the name table type is some sort of variable as FIX, FLOAT, or CS then a pointer is put in the PB list pointing to the name table entry corresponding to the name in

the PF list. (Note: the name is not carried into the PB list. The name and value are both obtainable from the name table entry.) The element is then deleted from the PF list. If the type is PRIMITIVE an immediate branch to the primitive location in memory occurs. (Note: A primitive is not treated as a flowchart. No markers are placed in the PF or PB lists, nor is a PSW created.) Since post-fix notation is used for the operators, the operands that a primitive needs have already been placed in the PB list and the primitive can access these by using the constant areas common to all routines. If the type is FLCHDISK it is necessary to access the flowchart table from the disk, create a flowchart list and then proceed as described below for the type FLCHLIST. A FLCHLIST type calls for the creation of a new PSW, and the marking of the two execution lists, and then the execution of the flowchart invoked. The passing of parameters has not been implemented as yet.

The PB list for the example in the description of the PF list would look as follows after the scan for the variables C and D and just before executing the primitive ADD:



The following pages are a list of system routines and primitives which have been coded and are being or have been tested. The x's on the right hand side indicate which are primitives and which have been tested.

S. Wilkins

System Routines and Primitives

```
                                                                PRIMITIVE TESTED
*ADDFREE   INSERTS FREE ELEMENT INTO FREE LIST
*          ELEMADDR CONTAINS FREE ELEMENT ADDRX     *E-----*
*          ELEMENT INITIALIZED TO EMPTY              *LENGTH*
*          LENGTH IS IN SECOND WORD                  *      *
*          NO RETURN CODE
*
*ADD        ADDS 2 ELEMENTS TOGETHER                                X X
*           THE ELEMENTS MAY BE SYMBOLIC REFERENCES OR ACTUAL VALUES
*           THE ELEMENTS ARE RECEIVED IN THE POINTER-BACK LIST
*           CAN ADD FIXED OR FLOATING POINT NOS. IF TYPES ARE MIXED
*             CONVERTS TO FLOATING POINT
*           CAN CONCATENATE 2 CHARACTER STRINGS
*           PRIMITIVE CALLED BY 'ELEM ELEM ADD'
*
*CHRCV1     CONVERTS FLOATING POINT NUMBER TO CHARACTER STRING
*           RECEIVES ADDRESS OF NUMBER IN REG 2
*           RECEIVES ADDRESS OF OUTPUT AREA IN REG3
*           LENGTH OF OUTPUT IS 20 CHARACTERS WITH 7 DECIMAL PLACES
*
*CHRCV2     CONVERTS FLOATING POINT NUMBERT TO CHARACTER STRING WITH
*           'D' FORMAT
*           RECEIVES ADDRESS OF NUMBER IN REG 2
*           RECEIVES ADDRESS OF OUTPUT AREA IN REG 3
*
*CREATE    CREATES ELEMENT OF REQUESTED SIZE FROM FREE LIST            X
*          ELEMSIZE CONTAINS TOTAL LENGTH OF ELEMENT NEEDED
*          RETURNS 32 BIT ADDRX IN ELEMADDR
*          DOES NOT INITIALIZE ELEMENT IN ANY WAY
*
*CRFLCTBL CREATES FLOWCHART TABLE  IN CONSTANT AREA                    X
*          FLAGS TABLE  ENTRIES ACCORDING TO ARROWS TO AND FROM
*            BIT 0 - UNUSED AT THIS STAGE
*                1 - SINGLE ARROW FROM
*                2 - MULTIPLE ARROWS FROM (BIT 1 ALSO ON )
*                3 - ARROW TO
*          WRITES TABLE ON DISK (NOT IMPLEMENTED)
*          PUTS ENTRY NAMES IN NAME TABLE
*          IF ERROR OCCURS IN TRANSMISSION REQUESTS RETRANSMISSION
*
*DISPLAY1 USES SNAP MACRO TO DISPLAY1 CORE BETWEEN 2 LOCATIONS         X
*          CALL DISPLAY1,(ADD1,ADD2)
*          ADD1 AND ADD2 ARE BEGINNING AND ENDING ADDRX-FULL WORK BNDRY
*
*DISPLAY2 PRINTS OUT A LIST OF LINKED ELEMENTS                         X
*          CALL DISPLAY2,(ADDRX)
*          ADDRX IS ADDRESS OF 1ST ELEMENT IN LIST
*          PRINTS OUT LIST ELEMENTS UNTIL REACHES ELEMENT WITH NO
*            RIGHT POINTER
*
*DVD      DIVIDES ELEMENT1 BY ELEMENT2
*           THE ELEMENTS MAY BE SYMBOLIC REFERENCES OR ACTUAL VALUES  X
*           THE ELEMENTS ARE RECEIVED IN THE POINTER-BACK LIST
*           CAN DIVIDE FIXED OR FLOATING POINT NOS. IF TYPES ARE MIXED
*             CONVERTS TO FLOATING POINT
*           NO CHARACTER STRINGS ARE ALLOWED
```

```
                                                                          PRIMI-  TEST-
                                                                          TIVE    ED
*             PRIMITIVE CALLED BY 'ELEM1 ELEM2 DVD'
*
*EXFCUTOR    EXECTS FLOWCHART IN LIST FORM                                        X
*            RECEIVES 1ST EXECUTABLE SYMBOL IN CURRELM
*            CAN  CALL ITSELF AS MANY TIMES AS NECESSARY
*
*FLCCNV      CONVERTS CHARACTER STRING TO DOUBLE PRECISION FLOAT PT NO.           X
*            REG 0 - PARAMETER ADDRESS 08WW0000
*              WW IS HEX LENGTH OF STRING
*            REG3 - ADDRESS OF CHARACTER STRING
*            REG 2 - ADDRESS OF DOUBLE WORD FOR FL PT NO.
*            RETURN CODE IN REG  15
*              0 = CONVERSION SUCCESSFUL
*              4 = INVALID CHARACTER STRING
*
*LISTREP     CREATES FLOWCHART REPRESENTATION FROM FLOWCHART TABLE                X
*            CREATES FIXED POINT AND CHARACTER STRING ELEMENT FOR EACH
*              TABLE ENTRY
*            LINKS SYMBOLS TOGETHER AND TO NAME TABLE
*
*MPY          MULTIPLIES 2 ELEMENTS TOGETHER                              X
*             THE ELEMENTS MAY BE SYMBOLIC REFERENCES OR ACTUAL VALUES
*             THE ELEMENTS ARE RECEIVED IN THE POINTER-BACK LIST
*             CAN MULTIPLY FIXED OR FLOATING POINT NOS BUT ALL RESULTS
*               ARE FLOATING POINT
*             NO CHARACTER STRINGS ARE ALLOWED
*             PRIMITIVE CALLED BY 'ELEM ELEM MPY'
*
*PRINT        DISPLAYS VALUE OF ELEMENT REQUESTED                         X
*             FORM OF OUTPUT IS 'NAME = VALUE'
*             PRIMITIVE CALLED BY 'ELEM PRINT'
*
*RECOGNZ     BREAKS CS ELEMENT INTO COMPONENTS                                    X
*            ELEMADDR CONTAINS ADDRESS OF CS ELEMENT
*            PUTS LIST STRUCTURE INTO PFLIST
*            SETS PFORWARD TO BEGINNING OF LIST
*
*SET          CREATES AN ENTRY IN CURRENT NAMETABLE                       X       X
*             RECEIVES 3 PARAMETERS IN THE POINTER-BACK LIST
*               1. NAME OF ELEMENT TO BE CREATED
*                  ALWAYS CREATES NEW ELEMENT EVEN IF NAME PREVIOUSLY USED
*                  SUBSEQUENT REFERENCES TO NAME ARE TO LATEST
*               2. TYPE OF VALUE THE NAME IS TO BE ASSIGNED
*                  FIX---FIXED POINT NO
*                  FLOAT-FLOAT POINT NO
*                  CS----CHARACTER STRING
*               3. VALUE TO BE ASSIGNED TO NAME
*                  PRESENTLY THE VALUE TYPE MUST BE IDENTICAL TO TYPE
*                  REQUESTED IN PARAMETER 2
*             THIS PRIMITIVE IS CALLED BY 'NAME TYPE VALUE SET'
*
*SETUP       PREPARES A FLOWCHART LIST FOR EXECUTION                              X
*            RECEIVES NAME TABLE ENTRY FOR FLOWCHART IN ELEMADDR
*            CREATES PSW ENTRY FOR FLOWCHART
*            ADDS ENTRIES TO NAMETABLE IF ANY
```

		PRIMITIVE	TESTED
*	MARKS PB AND PF LISTS RETURNS 1ST EXECUTABLE SYMBOL ELEMENT ADDRESS IN CURRELM		
*SUBT	SUBTRACTS ELEMENT2 FROM ELEMENT1 THE ELEMENTS MAY BE SYMBOLIC REFERENCES OR ACTUAL VALUES THE ELEMENTS ARE RECEIVED IN THE POINTER-BACK LIST CAN SUBTRACT FIXED OR FLOATING POINT NOS. IF TYPES ARE MIXED, CONVERTS TO FLOATING POINT NO CHARACTER STRINGS ARE ALLOWED PRIMITIVE CALLED BY 'ELEM1 ELEM2 SUBT'	X	X
*TRANSEBC	TRANSLATES 6-BIT CHARACTER CØDE TRANSMITTED BY PDP-7 TØ 8-BIT S/360 CØDE RECEIVES PARAMETER ADDRESS IN REG 1 ØF A FULL WØRD CØNTAINING THE LENGTH OF THE 6-BIT CHARACTER STRING. THE CHARACTER STRING IS LØCATED IN MEMØRY IMMEDIATELY FØLLOWING THE LENGTH. THE RESULTANT CØDE ØVERLAYS THE ØRIGINAL		X
*TRANSPCP	TRANSLATES 8-BIT S/360 CHARACTER CØDE TØ 6-BIT CHARAC- TER CØDE FØR TRANSMISSIØN TØ PDP-7 RECEIVES PARAMETER ADDRESS IN REG1 ØF A FULL WØRD CØNTAINING THE LENGTH ØF THE 8-BIT CHARACTER STRING THE CHARACTER STRING IS LØCATED IN MEMØRY IMMEDIATELY FØLLØWING THE LENGTH THE RESULTANT CØDE ØVERLAYS THE ØRIGINAL CØDE		X

3.4 Hardware Modifications

Final hardware modifications were performed on the PDP-7/2701 channel. The channel is operational and running test programs.

Final testing has been completed on the TTY receiver and transmitter module. The logic is being utilized in the full duplex mode and appears to be very reliable. The transmitter and receiver were realized from TTL logic and are compatible with our normal I.C. logic levels. The unit is certainly fast enough for high speed data line use. The cost of the complete receiver, transmitter and clock will be about $120 per port.

The display drive hardware has been mounted and the x and y registers checked out and tied to their respective converters. Decoder logic has been checked out as far as possible, pending a solution for furnishing sequencing clocks.

The disk to PDP-8 interface of integrated circuits was finalized, design checked and integrated circuit portion wired and checked. The cabling, wiring, and level conversion has been done. The IOT boards were cut and installed to furnish I/O instructions and gating to the interface. A change has been made in the PDP-8 master clock area to change the PDP-8 cycle time for disk SYNC purposes. During disk transfers the PDP-8 clock is furnished by the disk and the PDP-8 is forced to run a two-usec cycle time in lieu of its normal 1.5 usec cycle time.

At the present time there is not a facility for interrupt programming on disk transfers. This means for all practical purposes a maximum of 34 msec may be wasted when a disk block transfer is requested. This is temporary, and was done primarily to facilitate checkout.

Disk usage will consist of transferring two 12 bit words from the AC to the disk interface. One control word, one address word. The transfer is accomplished with a particular IOT instruction. Block transfers of 1264 words or less only are available.

Disk instructions at present are:

IoT 6551 Gate AC to disk address buffer. (Address is starting address of PDP-8 memory block).

IoT 6561 Gate AC to disk control rec. (Selects disk sector, track selection, transfer direction, PDP-8 mem. extended address bits).

IoT 6552 CLR disk done

IoT 6562 Skip on disk done

3.4.1 Display Terminal

The disk controller which is a part of the display terminal has been designed and almost completed. The disk we are using is manufactured by Data Disc, Inc., and has a fixed head for each of the 64 tracks. The controller handles data in blocks of 1365 12-bit words by using the PDP-8 data break facilities.

Disk data rate per track is 3.0 Mbps. In order to transfer a 12-bit word in approximately 2uS, data is transferred through two channels, fed by two shift registers, 6 bits each. Parallel data transfer to and from the shift register for both the write and read cases is determined by the disk clock which is modified by dividing its pulse rate by 3. In order to synchronize the disk and memory data rates, the modified disk clock replaces the PDP-8 clock during data break time. The disk interface can directly address the total PDP-8 memory.

The designated memory address should be loaded into the accumulator and transferred to the disk interface with an IOT instruction. Before initiating the disk transfer by a second IOT instruction, and accumulator should be loaded again with group and sector addresses and transfer direction.

Work has begun on the design of the display central processor. We are aiming for an inexpensive general purpose remote display terminal. The processor will decode the computer data word and activate the appropriate function generators. The display generator may include such function generators as a character generator, vector (line) generator, a circle

generator, position generator, and a dot generator. Because of the need to refresh the displayed image 30 times per second, it is advisable that the function generator together with the refresh storage (disk in our case) be part of the CPU. This disk will be used to refresh the display without interrupting other computations.

Efforts were directed at finding the best way to form lines and characters. In the incremental deflection,on-dot method lines are formed by closely spaced dots. These dots are centered on points on a quantized field. Characters can be formed by a matrix of dots. Character dot information can be supplied by separate information words. Another way is the stroke method in which characters are drawn as a sequence of short lines. The memory supplies the orientation of each of the lines. Normally, eight possible orientations are stored and nine to twenty line segments are used to display a character.

The current display drive has been built to draw lines by a series of points or dots. Thus, the most compatible scheme for displaying characters is a rectangular array of dots. The simplest array that will still give the needed readability for all characters is an array of 35 dots--seven dots high, five dots wide. All characters can be made up from such an array by selectively intensifying specific points.

"Read Only" memory may serve as a storage for character-data (96 characters). The desired character is selected and the 35-bit word containing the intensity information of the character will be transferred from the ROM to the disk. All other characters or other functions to be displayed will be stored on the disk.

The display cycle will start by reading this data from the disk into the display registers and repeating this process for refreshing the displayed image.

C. Carter
R. Hostovsky
T. Kerkering
H. Lopeman
R. Miller

3.5 PLORTS Group

ALBERT, the PLORTS PDP-7 program, has been modified and extended. The ASP 1.5 system has been replaced by ASP 2.0; a number of modifications for PLORTS have been made thereunto. The Phase 1 RJE facility (one terminal) has been implemented. The filing system for Phase 0 and beyond has been defined again, as have the requirements for Phase 0 RJE. Work on the first two passes of the compiler/executor has continued; attention has centered on Pass III.

3.5.1 PDP-7 Assembler

A PDP-7 assembler has been completed which runs on the 360, and has been added to that system. It was written to supplant the Illiac II version, as the Illiac has been removed from service and disconnected.

The 360 version has several advantages. For the first time it is possible to utilize card input, and obtain a listing some 500 times faster from a high speed 1403 printer instead of a teletype. Character pseudo instructions have been added to allow defining of messages as character strings rather than as absolute octal constants. A sorted symbol table and reference table have been added, which have proved quite useful. The standard expression analyzer for address calculations allows only PLUS, MINUS, AND, and OR as operators. This has been expanded to allow nested parenthesization, and any arithmetic or boolean operators. Also allowed are type qualifiers in expressions to allow specification of decimal or hexadecimal numbers and the bit representation of an ASCll character.

The only useful output from such an assembler is paper tape. For the present, this is being punched via an IBM 1800 which is attached to our 360. With the installation of ASP 2.0 appropriate DSP's (Dynamic Support Programs) will be included so that this output may be directed to the PDP-7's much more satisfactory paper tape punch. The ability to punch paper tape from the 360 will mainly be used as a service for 360 time-sharing users.

Although the single PDP-7 program to handle timesharing consoles warrants a good assembler, it is speculated that far greater use of the assembler will come about when it is realized that it may be used for PDP-8 programs.

3.5.2 ALBERT

ALBERT, the PDP-7 program for handling timesharing terminals, has been modified for use on the 360. The new assembler options were utilized in places where they were badly needed. Some operator functions were added through the on-line console to allow checking of the state of the PDP-7 during normal operations. From this console we can now send messages (zB TS ON) to the terminals and inquire about system activity--number of lines to and from the 360, I/O errors, usw.

3.5.3 Communication Link

The hardware connection between the PDP-7 and the 360/50 is now functioning properly.

Two ASP 2 DSP's, PDPIN and PDPOUT, have been written and presently reside in a test version of the ASP system.

PDPIN is the one-terminal RJE program which was being tested under ASP 1.5 at the end of last quarter. There are no known problems in PDPIN--it will be added to the working ASP system as soon as a few minor improvements are made. Under the test ASP system, we are able to enter jobs from a terminal into the ASP input stream with no apparent difficulties or restrictions.

PDPOUT is a DSP to send output from a job run under ASP to a terminal or to the PDP-7 paper-tape punch. PDPOUT is independent of the input source for the job; it may be called by a job submitted through the "normal" (card) input stream as well as by a PDPIN job. PDPOUT has successfully transmitted to a terminal; paper tape output has not yet been tested.

3.5.4 ASP

The conversion to ASP 2 was effected in September. The working version of ASP 2.0 includes modifications parallel to those previously discussed for ASP 1.5; these constitute sufficient support for PDPIN and PDPOUT, and in fact, for any program using the ASP partition (partition 0) for PDP-7 I/O.

A test version of ASP 2.0 which will enable any partition to handle PDP-7 I/O and then return control of the PDP-7 to ASP, has been written and found to function with the PDPIN/PDPOUT DSP's (d.h., it is no worse than the working system). I/O from partition 1 has not yet been executed successfully.

Interpartition communication requires minor redefinition of The Use and Need* of two features of ASP: the VIOLATE macro and the TVTABLE. The former links to an ASP SVC which allows the user to execute a piece of his own program in supervisor state and under zero storage-protect key. In particular, it allows data to be stored into a different partition or into the OS nucleus.

The latter, a transfer-vector table pointing to relevant entry points scattered about ASP, has been extended to include, among other goodies, pointers to tell the PDP-7 attention-interrupt routine the locations of the appropriate TCB and ECB for posting the interrupt. A partition 1 program may set its own addresses in these slots, carry out its allotted functions, restore the ASP addresses in the table, and permit the ASP partition to regain control of the PDP-7. The address of TVTABLE is made available to other partitions via a bit of sleight-of-core. The ASP initialization routine alters the SVC new PSW to point to the SVC-filter entry point in ASPCONTL. We have simply inserted just ahead of this entry point a fullword pointer to the beginning of TVTABLE.

The STYX routine to gather performance statistics for ASP has not yet been converted to ASP 2.

---

* c.f., C. A. Nation

3.5.5 Filing System

The current final definition of the PLORTS filing system involves the writing of a Basic Line-Numbered Access Method. The (disk) files are to be stored in compressed format in 900-byte records. Access to these files for programs on Main(the 360/75) will be restricted to the use of the ASP /*DATASET feature and to a read-only subset of the Basic Line-Numbered Access Method.

PLORTS will have some 8-10 entry points to BLAM for performing its divers functions. Linkage in all cases will be in essentially the following format:

```
BAL     14,BLAM
DC      AL1(CODE)     FUNCTIØN TØ BE PERFØRMED
DC      AL3(NAME)     FILE NAME ØR USER ID PTR
DC      AL1(BLEN)     LNTH ØF CALLER'S BUFFER
DC      AL3(BADD)     ADDR ØF CALLER'S BUFFER
```

Any data (other than the file name) which must be passed to BLAM is in the buffer at call time. In particular, a read/write call will have the line number at the beginning of the buffer. The notion of a line number has been completely generalized. A catalog (directory) entry, for instance, is handled the same way as a data record; the name of the target file is at the beginning of the record and is treated by BLAM as a line number.

Upon return to the caller, BLAM will have taken whatever action is appropriate and possible (zB, read the requested record into the caller's buffer), and will have set up a code byte indicating to the caller what error conditions, if any or at all, were encountered.

An outline of the BLAM functions is found at the end of this section.

3.5.6 RJE Facility

As noted above, remote job entry for one terminal at a time already exists. For live Phase Zero RJE, the user will enter a command such as

```
RUN  FILE1,FILE2,FILE1469
```

The executor will read the named user files (via BLAM), and will pass to the ASP partition a sequence of card images representing the concatenation of the specified files (typically one or more JCL files and a source program).

Within ASP there is a DSP, the function of which is to "read" these records from partition 1 and pass them to ASP input service for processing. This DSP is logically parallel to the RDR or PDPIN DSP's.

Basic Line-Numbered Access Method Routines

Initialize

Returns: Success
System error
Catalog not found
Buffer too short

It is assumed that there is no valid file data in swappable core. All pointers are reset and control is passed to the Search routine to look up the user's catalog using his name and problem spec number.

Search

Returns: Success
System error
Catalog not found
Buffer too short

Search looks up the supplied name in the current catalog and places the catalog entry data in the caller's buffer. If the file whose name was supplied is itself a catalog, it becomes the new (current) user catalog.

Open

Returns: Success
System error
New file created
Catalog
Disk full
Already open
No core for FCB

The user catalog is searched for the named file. If it is found, a File Control Block is constructed. If it is not found, a new file is created and an FCB constructed for it.

Opensys

Returns: Success
System error
Not found
Catalog
Already open
No core for FCB

This allows one user to open another user's file as a read-only file. The main directory and requisite subsidiary directories are searched until the named file is found and an FCB is constructed for it.

Close

Returns: Success
System error
Not open

Close destroys the FCB created by Open or Opensys. The current user catalog may not be closed.

Read

Returns: Success
System error
Record not found
End of file
Buffer too short
File not open

Read locates the specified record in the named file and places it in the caller's buffer. If the specified record is not found, the next higher record in sequence is returned.

Readseq

Returns: Success
System error
End of file
Buffer too short
File not open

Readseq returns the record following the last one read. Otherwise, it behaves like Read.

Write

Returns: Success
System error
File not open
New record
Disk full

Write locates the specified record in the file and replaces it with the caller's record. If no match is found, the new record is inserted in the file according to its line number. If the named file is a catalog and a new record is being added, a file is created with the corresponding name.

Terminate

Returns: Success
System error

Terminate closes all files currently open for the user.

3.5.7 Pass I

The arithmetic expression and assignment statement routines now have no known bugs. These routines are: CØNSTANT, EXPRESSN, ØPERATØR, EXPSNSQ, IDENTIFYR, TPPPHØP, ASSIGNST, PØPBØSXI, GETNBLR, PUSHBØS, and LHSIDE. (EXPSNSQ handles subscripts, TPPPHØP handles stack manipulations involving operators, PØPBØSXI handles the popping of beginning-of-expression markers, and PUSHBØS pushes them). Numerous modifications have been made to these routines. Asterisk subscripts are now handled properly. LHSIDE now flags as a fatal error the occurrence of a subscripted label with a non-constant subscript. Multiple assignment is now recognized and rejected.

Call-by-value is now handled properly. If (<unit>) occurs in a subscript list on the right-hand side of an assignment statement, the left parenthesis causes the "possible-call-by-value" flag to be set. The routine EXPRESSN syntax checks <unit> and if it is other than a constant, identifier, or subscripted identifier, the "not-call-by-value" flag is set. When the right parenthesis is recognized, both flags are checked; if the "possible-call-by-value" flag is set, and the "not-call-by-value" flag is not set, a temporary operator symbol-table pointer is output. The "possible-call-

by-value" flag is stacked in the recursive storage for EXPSNSQ, allowing the call-by-value analysis to be independent of the nesting of subscripts (e.g., 'A((B((C((1))))))' will cause three temporary operators to be generated).

The routines PA, INØUT, and DEC which handle the PACKED and ALIGNED, INPUT, ØUTPUT and FILE, and DECIMAL attributes respectively are now debugged in addition to those previously debugged.

All string delimenters output by attribute routines are now output in halfword symbol-table pointer form. Dummy operators for colons and replication ranges (in the INITIAL statement) have been devised.

The INITIAL attribute analysis routine has been revised; LABEL arrays are not allowed to be initialized using the INITIAL statement, and if a replication factor is seen, left and right dummy operators are output to delimit the range of the replication factor. This will allow the initialization to be executed properly in Pass III.

The DECLARE statement package has now geen completely debugged. The INITIAL and DEFINED attribute syntax analyze routines now work, as does the DCLST analysis of grouped attribute constructions. The ENTRY and RETURNS attributes have been excluded for the initial implementation. The final format for the secondary pointer for further attribute information is as follows:

BYTE 0-2 pointer to beginning of string for DEFINED attribute
2-3 pointer for INITIAL attribute
4-5 pointer for BIT attribute

A colon operator equivalent to the colon separating the lower and upper bounds of an array subscript has been incorporated in the symbol table.

KEYWORD has been modified so that it now has two external entry points, KEYWORD and KEYWORD1. An entry to KEYWORD implies that a keyword has not been seen and IDENTFYR is called to check for a legal keyword; an entry to KEYWORD1 implies that the keyword has already been seen and resides

in IDENTBUF, and the routine goes directly into the comparison against the keyword dictionary. The method of loading the address of the section of the dictionary to be searched has been modified to optimize the object code.

All the separate statement syntax checking routines have been updated and modified to incorporate recent revisions and new linkage conventions. The whole of Pass I except for line-number checking and statement-buffer garbage collection is now in the debugging phase.

The convention on the use of R7 is as follows. All subroutines assume that R7 is an index except LHSIDE, ASSIGNST, and the arithmetic expression analysis routines, which assume that R7 contains the address of the buffer to be output into. When LHSIDE detects a statement which cannot be an assignment statement, R7 is restored to index function.

LHSIDE has been revised. If a character string is seen which is suspected by character count to be a keyword, the character following it is scanned. If it is a colon or an equals sign the input pointer is reset to re-analyze the suspected keyword as an identifier of too many characters.

If a left parenthesis is seen, the routine counts subsequent left and right parentheses until the counts are equal, or until the end of the line is reached. The latter case is a fatal error. In the former case, if the character after the nest of parentheses is a colon or an equals sign the input pointer is reset to reread the subscripted identifier as an identifier of too many characters (see above). If it is anything else, the input pointer is also reset, but control transfers to the KEYWORD routine and a keyword is assumed. The section of code for doing this is being debugged currently.

GETNBLCR and LINGND1 have been modified so that the latter uses the former to analyze characters following the semicolon. Comments or blanks are read and ignored; any other character is ignored but a warning to the effect that only one statement or clause is allowed per line is issued.

The PRØCEDURE statement syntax analysis has been modified to detect whether associated labels are subscripted, and if so, an error is flagged.

The FREE statement routine now recognizes the pointer-variable construction and rejects it since this is not implemented in PLORTS.

3.5.8 Pass II

Two utility routines, SST and SCB, have been written. The former has two parameters: R3 points to an identifier, R2 points to the present high location in the symbol table. This routine searches all sections of the symbol table which contain names already declared in the given block and all containing blocks for the first occurrence of the identifier pointed to. The search is bottom-up, i.e. the block containing the name is searched first then the next block in which it is contained, etc. The search utilizes the block table. Closed blocks are skipped and DØ groups are ignored. If the pointed-to identifier is not found, a fail exit is taken; if it is found, the pointer to it is returned in R2.

The parameters and conventions for SCB are like those of SST, the difference being that SCB searches only through the block being currently analyzed.

The subroutine LABELS, which handles all labels, is being written. At present a method is being devised to handle subscripted labels, i.e. determine whether or not they are in range and place the proper address in the correct place in storage reserved for that array. Given a label array

$$A(S_1:S'_1:S_2:S'_2,\ldots,S_n:S'_n)$$

S_i, S'_i are integer constants denoting the lower and upper bounds of the given subscript respectively. If a subscripted label of the form

$$A(x_1,\ldots,x_m)$$

is seen (x_i integer constants), n is compared with m. $n \neq m$ is a fatal error.

If n=m then each of the m inequalities $S_i \leq x_i \leq S_i'$ is checked. If any are not true a fatal error occurs. If all are true the position of

$$A(x_i, \ldots, x_m)$$

is calculated by the formula

$$(x_1-S_1) \prod_{i=2}^{n} (S_i'-S_i+1) + (x_2-S_2) \prod_{i=3}^{n} (S_i'-S_i+1) + \cdots + (x_n-S_n)$$

which gives the proper index in the array of $A(x_1, \ldots, x_n)$. The address of the statement labeled with this label in the Pass II output string is placed in the location so calculated.

The Pass II PRØCEDURE statement analysis has been modified. It now correctly handles the case of a procedure with the attribute of BIT (expression) where some of the identifiers in the expression are possibly undefined at that point in the program. Those names are entered as EXTERNAL names in a section of the symbol table set aside for them. These are limited presently to 100 in number.

The Pass II ENTRY statement routine has been written. It is basically the analysis used in the PRØCEDURE statement analysis with these differences. All labels on the error statement are stacked temporarily in a special table, which upon reaching the END statement for the block, is transferred to the symbol table (SYMTAB) section for the containing block. No block table entry is made, the declarations internal to the block are searched for parameters. If the parameters are not found, symbol table entries are made for them.

The BEGIN statement Pass II analysis has been coded. It calls a routine (as yet unwritten) which processes labels and makes a block table entry.

A preliminary flowchart for the Pass II DECLARE statement has been written but it is still in rough form.

The symbol table format has been altered. A provision has now been made to include information about the storage type and reserve bits for label variables and file names.

This format is as shown on the SYMBØL TABLE.

SYMBØL TABLE

BYTE 0

bits 0,1	0	constant
	1	variable
	2	temporary (in PASS II statement label constants)
	3	{bits 2-7 off--dummy parameter bit 7 on--operator
bit 2,3	0	character string
	1	bit string
	2	bin fix
	3	bin float
4	0	single precision
	1	double precision
5	0	real
	1	complex
6	0	nondimensioned
	1	dimensioned
7	0	not assigned
	1	assigned

BYTE 1

(Character string or bit string) Length 1

bit 0 (if=1)		parameter
1 (if=1)		*subscript
2 (if=1)		label variable
3,4	2	output file
	0	input file
	3	print file
5,6	0	automatic
	1	static
	2	controlled

BYTE 2→3 pointer to location in core (PASS III)

A character table (CHARTAB) entry corresponds to each symbol table entry and consists of 8 bytes.

3.5.9 Pass III

Execution-time code is in the form of "polish postfix." Polish statements are all of one type, consisting of a string of polish words (equivalent to 360 halfwords) which are displacements for a symbol table. The symbol table entry identifies the polish word as one of the following:

1 - constant
2 - variable
3 - temporary result
4 - star subscript
5 - function (either user or system)
6 - operator

During execution the symbol table entry is put into an operand stack unless it is an operator, in which case the operation is performed. All operations are performed on the stack. Individual PL1 statements are differentiated not by separate statement types, but by the nature of the operators involved.

The operators currently in use are as listed in Table 1.* Most are for normal arithmetic expressions and need no special explanation. The following example will illustrate the general form of an assignment statement and a special feature of the assignment equals operator:

A,B=G*F

The above statement is recoded by previous passes into polish postfix as follows:

ABGF*==

The first operation performed is multiplication, which removes pointers to G and F from the operand stack, and returns a single pointer to the result. The equals operator removes pointers to B and the product of G and F and performs a move operation, after which it returns the result pointer to the stack, permitting multiple assignment as shown.

In order to demonstrate some of the less obvious aspects of Pass III, each of the special operators will be explained separately.

---

* See end of this Section.

TEMPORARY GENERATOR

This operator was created to handle call by value for the following type of statement:

```
CALL FUNC((B))
```

PL1 rules state that the variable B must be copied into a temporary location so that storing into the parameter by the subroutine will not affect the value of B.

END OF STATEMENT

The "semicolon" operator is used to clear the operand stack of garbage (assignment equals always leaves an extra entry) and tests to see if the statement needs to be re-executed due to the occurrence of matrix operations. A table of six counters is maintained (for each of the six possible subscripts allowed in this PL1 subset) which serve as counters for repeated execution of the statement and as a reference to the non-specified subscript values to be used in a given pass.

For example, the following arithmetic statement must be executed five times:

```
DCL A(5), B(5);
A=B(*)+1;
```

One counter is involved for the one variable subscript, with a range of one to five. This subscript will be substituted for the missing subscripts for A and B during each execution of the statement.

Although the semicolon operator increments the subscript counters and determines the number of times to execute the statement, it has nothing to do with originally determining that this is required. Each of the starred operators in Table I examines the operands it obtains from the stack and determines whether or not they are dimensioned (a single bit in each symbol table entry specifies this). The A in the above example will be caught in this manner when the assignment equal operator attempts to do a move operation. The operator will hand the symbol table entry to a Pass III routine named Element which returns a faked symbol table entry, undimensioned, which points to one of the elements of the array it was given.

The Element routine obtains the pointer to an array element in one of two ways. If subscript counters already exist, the limits to which the subscript counters will be incremented are checked against the subscript limits of the array in question. These and the number of subscripts must agree or an error condition is raised and execution ceases. The subscript information for the array is contained in a dope vector immediately before the data for the array. If a symbol table entry is flagged as dimensioned, it is assumed that the pointer is to the dope vector rather than to the actual data.

If the array is determined to be correctly dimensioned, the subscript counters and the implicit length of a single element of the array (as determined from its type) allow construction of a symbol table pointer to one of its elements. If the subscript counters have not yet been established then the limits are copied from the dope vector and the counters are initialized to the lower bounds of the array. In this case the faked symbol table entry returned points to the first element of the array.

The subscript counters can be initialized in an additional manner. A subscript operator is used after an explicit declaration of subscripts by the programmer. To illustrate, the following PL/1 statement segment:

A(2,3,*)

would be recoded in polish as:

A23*3!

where ! is taken to be the subscript operator, and the preceding 3 to be the number of subscripts. In this case the required element of A is determined by the subscript operator before the array is accessed by an arithmetic operator. In this example the subscript operator will note the invocation of one subscript counter with the bounds associated with the third subscript that was declared when A was originally dimensioned. The subscript operator will handle missing starred subscripts the same way the Element routine handles them, except that the number of subscripts involved is taken to be one rather than three as this is the number unspecified.

RETURN AND RETURN WITH A VALUE

Return provides a means for returning from called procedures without passing control to the end statement. The return operator is intended for use with procedures invoked by a Call statement. When control is returned to the Call statement the next thing to be executed should logically be a semicolon operator, and the existenence of a semicolon operator is checked for before a return is made. This prevents the programmer from performing a return from a function procedure without the function's value. Similarly, return with a value specifically checks for an absence of the semicolon operator. It should be noted that the CALL statement is handled internally by the same mechanism used for function calls to maintain the concept of a single statement type in Pass III.

The following typical return statement:

RETURN (exp);

is coded into polish as folows:

(polish exp)#

where # is taken to be the return with a value operator.

LOAD POLISH LOCATION COUNTER

The Loac PC operator is used to implement the GO TO statement. It requires one operand: the polish branch address.

TEST AND LOAD PC

This operator performs the basic branching decision required for the IF statement. One operand is a one bit bit string whose value is supplied by the Boolean expression from the If statement. The second operand is a polish branch address which is to be loaded into the polish location counter if the value of the bit string is false. The following PL/1 statement:

IF A=B THEN (statement);

would be translated into:

AB=m$(statement);

where $ is the Test and load PC operator.

Small m is an internally generated branch address pointing immediately after the THEN clause, whether this clause is a single statement or block. It must in addition point to a position following the code produced by a possible ELSE statement. In this way ELSE may be reduced to an unconditional GO TO statement.

BUILT IN MATHEMATICAL FUNCTIONS

All built in functions such as SIN, COS, ATAN, etc., require one operand and return one operand. According to the PL/1 reference manual, if the argument for one of these functions is an array, the function must return an array as a result. However, since PL/1 statements involving matrix operations are executed in entirety for each separate element of the arrays involved, the built in functions are handled as operators rather than functions. This is so that they may call the Element routine to obtain the element from the array which is to be involved in a given pass through the statement.

INPUT/OUTPUT OPERATORS

Four operators are currently implemented. They are PUT EDIT, GET EDIT, PUT LIST, and GET LIST. Input/output statements are broken down into several separate statements for execution. LIST-directed transmission operators require one operand: the variable or expression involved in the input/output operation. A separate PL/1 statement is generated for each variable to be printed or read. Prior to these statements is a function call to a system subroutine to enter the name of the file or string to be involved in the transmission.

To illustrate the use of EDIT directed transmission the following PL/1 statement:

```
PUT EDIT (A,I DO I=1 TO 5)
     (F(20,I),F(4));
```

would appear to Pass III as though it were the following sequence of statements (before coding in polish):

```
CALL SYSFML (m);
CALL SYSFN (SYSPRINT);
DØ I = 1 TO 5;
   PUT EDIT (A);
   PUT EDIT (I);
END;
   .
   .
   .
m: F(20,I),F(4) GO TO m;
```

Small m is an internally generated branch address for the format statement. The system subroutine, SYSFML, hands the input/output routine this polish branch address; when the PUT EDIT statements are encountered, control will actually be passed to the format statement (m) which will have been coded into polish like any other statement. Execution will proceed inside the format statement up to and including the first data requiring operator. When control leaves the format statement SYSFML will retain the last execution address. The manner in which the format statement is coded into polish will be discussed in the next section. The call to SYSFN is to enter the name of the file or string involved in the transmission.

FORMAT OPERATORS

A class of format operators has been provided so that the format statement may be coded into polish. A PUT EDIT or GET EDIT operator requires one operand, but merely passes it on for use in the format statement. Data requiring format operators is preceded by sufficient operands to specify the necessary field width and precision. The operator removes from the stack these parameters and the operand left over from the PUT EDIT operation.

To illustrate, the following format item:

F(20,I)

would be coded as:

20 I F

The F operator would remove three operands, thereby obtaining the necessary data element as well as the field parameters.

TABLE I

OPERATOR LIST

OPERATOR		OPERANDS REQUIRED	RETURNED
*Multiply		2	1
*Subtract		2	1
*Add		2	1
*Divide		2	1
*Exponentiate		2	1
*Negate (Unary minus)		1	1
*Or		2	1
*And		2	1
=	(Comparison)	2	1
<	(Comparison)	2	1
>	(Comparison)	2	1
<=	(Comparison)	2	1
>=	(Comparison)	2	1
¬=	(Comparison)	2	1
¬<	(Comparison)	2	1
¬>	(Comparison)	2	1
*= (Assignment)		2	1
*Temporary generator		1	1
End of statement (Semicolon)		any remaining	0
*Concatenate		2	1
*Not		1	1
Return		0	0
Return with value		1	1
Load PC		1	0
Test and load PC		2	0
*Built in mathematical functions (sin, cos, atan, etc.)		1	1
*Put edit		1	passed
*Get edit		1	passed
*Put list		1	passed
*Get list		1	passed
Format operators			
F(w,d)		3	0
E(w,d)		3	0
C		1	2
A(n)		2	0
A		1	0
B(n)		2	0
B		1	0
X(n)		1	0
Column(N)		1	0
Line(n)		1	0
Page		0	0
Skip(n)		1	0
Subscript/Function call		variable	1

The operators A and B are for output only and take their length parameter from the implicit length of the data.

The complex format item (C) duplicates the data item in the stack and modifies it so that the first item is the real part and the second item is the imaginary part. In addition, it sets a flag that format execution may proceed through two data-requiring operators.

SUBSCRIPT/FUNCTION CALL

This is a single operator which has the function of calculating array elements from their subscripts and handling function calls. Since a subscripted variable is difficult to separate from a function call, both are coded in exactly the same manner for Pass III. The use for calculating subscripts has been discussed previously.

Each operand loaded into the stack is inspected to determine if it is a function name. If it is, the previous portion of the statement is set to a quiescent state and reactivated by a RETURN statement. At this point the function call effectively becomes a separate statement. This is done so that the function call may be executed repeatedly in the event that matrix calculations are required within the parameters. In this use the subscript operator acts in a manner similar to that of the semicolon operator. The subscript operator determines the mode in which it is to operate when it examines the variable for which it is to calculate subscripts.

P. G. Boekhoff
J. L. Christopher
A. D. Whaley

3.6 Microprogramming Research

This report is primarily a compilation of the concepts, definitions, and problems which are under study. By putting them on paper now, we hope to simplify the production of later reports. Also, the principal progress of this period has been the identification and labeling of these concepts, so it seems only fair to include them at this time.

We start with a "facilities machine." This is nothing more than a set of registers, shifters, adders, random flip-flops, and the data paths connecting these "facilities." The data paths are controlled by gates of one sort or another (an AND circuit, for example), and each gate has one control signal. If the control signal's value is "1", then the gate is opened; if the value is "0", then the gate is closed. Due to the one-to-one relationship between the gates and their control signals, we can adopt a single set of symbols to represent them both without ambiguity. Henceforth, "g_i" will be the i-th control signal/gate; a g_i can assume the values of its control signal, 0 and 1.

This facilities machine can be viewed as a black box with two kinds of inputs and one kind of output. They are: data inputs, data outputs, and the $\{g_i\}$ as inputs. The $\{g_i\}$ are our "handles" on the facilities machine. By wiggling them we can cause data to be input into the machine, moved from register to register within the machine, logically manipulated, and output. Given the knowledge of each g_i's function, we can "program" the facilities machine by writing down a time-sequence of g_i activations, stating precisely when, in what order, and in what combinations the g_i should be set to "1".

Note that programming with the $\{g_i\}$ is somewhat tricky. There are no constraints concerning parallelism. All the $\{g_i\}$ may be set to "1" simultaneously if we wish. We must be careful that simultaneous activation of a set of $\{g_i\}$ does what we wish be done, but other than this functional

restriction we are free to employ as much parallelism in our programming as we want. Also, we are completely free to issue any combination of the $\{g_i\}$ at any time. Thus, there are no restrictions on either parallelism or sequentialism within the facilities machine.

In terms of the terminology of micorprogramming, the concepts introduced so far can be restated as follows. The "facilities machine" is the arithmetic/logical unit of a microprogrammed computer. The $\{g_i\}$ are called micro-orders, and a microinstruction specifies which of the micro-orders are to be activated during a single ALU cycle. The machine language instructions of the computer (like ADD, MPY, TRA, etc.) are implemented as subroutines of microinstructions; these routines are stored in the control memory or microstore, normally a high-speed read-only memory. Thus, a programming of the control memory really just defines an instruction set for the microprogrammed computer.

In this report we will stay with our original terminology. Thus, we will speak of programming the facilities machine and of instructions for the facilities machine (microprogrammings and microinstructions, respectively). This will be done to emphasize the generality of the problems under study, for these problems are inherent in any parallel machine. Future reports may lapse into the microprogramming jargon, however, so keep it in mind.

Consider now the problem of designing the format for an "instruction" for a facilities machine. A priori the problem is trivial. We just assign the i-th bit of the instruction to g_i. To execute such an instruction we just set to "1" all the g_i whose corresponding bit in the instruction is "1." This instruction format is just as general as the facilities machine itself. It permits as much parallelism as does the facilities machine, and although it appears to be restrictive in regard to the sequencing of the $\{g_i\}$, it is still true that any combination $\{g_i\}$ can be activated at any time. The practical realities of an instruction fetching system might restrict sequencing, but within the instruction the instruction concept itself there are no such constraints.

This solution is wasteful, however. It produces too wide an instruction. If there were N of the $\{g_i\}$ which we know can never be issued together meaningfully in any combination, then we could take $[\log_2(N+1)]$ bits of the instruction, assign the N g_i to the first N possible values these bits can take on, then decode these bits of an instruction together as a group to determine which of the N g_i is to be activated. This saves us N - $[\log_2(N+1)]$ bits. For example, if we had three such g_i, we could take 2 bits of the instruction and assign the four possible values as follows:

00 execute none of the three g_i
01 execute the first of the three g_i
10 execute the second of the three g_i
11 execute the third of the three g_i

A simple decoding network fed by these two bits could activate the desired g_i easily. The trick now is going to be to discover which of the $\{g_i\}$ can never be issued together meaningfully.

To simplify this part of the problem, assume that we are given a programming for the facilities machine. Such a programming would indicate precisely which pairs of the $\{g_i\}$ will be used simultaneously, and would avoid the problem of meaningful simultaneity all together. We assume that the programming is meaningful, so all parallelisms indicated are meaningful. Similarly, all unused parallelisms are not meaningful, at least relative to this programming they are not. Consequently, we can get on to other problems.

One interesting problem arises when instruction memory width is expensive but decoding trees of the type mentioned above are cheap. In this case we want to try to minimize the number of bits per instructions by grouping as many $\{g_i\}$ as possible together into fields of bits. Of course, we are going to probably have more than one field of bits (unless no parallelism was ever used in the given programming), but we want the largest possible fields in any case.

Although it is not strictly true, as a first approximation one can say that the fewer the number of fields used, the larger the fields, and thus the narrower the instruction. When the minimum instruction width problem is stated in these terms, its solution has been found to be equivalent to finding the chromatic number of a graph. The only known method of solution for this problem is linear programming. If there are N g_i it involves N^2 variables and N-times as many constraints as there are pairs of g_i occurring in parallel. To make this a practically solvable problem, tricks are needed, and trickology is currently being investigated, albeit leisurely. Having identified the computational nature of the problem, and found it formidable, we are content to move on to other problems.

The basic underlying problem of all this is minimization of the number of bits of memory necessary to hold a given programming of the facilities machine. Minimizing instruction width (given a programming) is one way. Another is to find an equivalent programming with either a) fewer instructions, or b) narrower instructions, or c) greater multiple use of instructions. Vague as it is, the third possibility is our current area of investigation; essentially, by adding conditional branching possibilities to a programming and by moving instances of g_i within the programming so as to maintain functional equivalence, how can we minimize the number of memory bits needed to express the given programming.

L. Greninger

4. ILLINOIS PATTERN RECOGNITION COMPUTER: ILLIAC III
(Supported in part by Contract AT(11-1)-1018 with the U.S. Atomic Energy Commission and the Advanced Research Projects Agency)

4.1 Introduction

Illiac III is an experimental computer being designed and constructed by the Department of Computer Science as a first instrument to explore the potentialities of high speed image processing. Besides providing normal computational facilities, the machine includes a parallel processor for visua data processing (the Pattern Articulation Unit) and an extensive complement of visual input/output equipment.

Also reported here are developments (funded by the Advanced Research Projects Agency) which attempt to provide picture processing facilities for remote users with only moderate image-processing demands. The strategy here is to extend the central processing facility (Illiac III) via a video communications net.

4.1 Outline of Illiac III Programming Developments

The section numbers of the outline below correspond to section headings of this Progress Report.

4.2 Inter-Machine Links

4.2.1 The Illiac III/PDP-8 Interface

4.3 Operating System

4.3.1 Job Control Language

4.3.2 Telecommunications Processing Package (TP^2)

4.3.3 Segment Linkage

4.4 Translators

4.4.1 IBAL (Assembler Language)

4.4.2 PL/1

4.5 Experimental Recognition Procedures

4.5.1 Recognition of Graph Transformation Grammars

4.5.2 Recognition of Context Free Grammars

4.5.3 Recognition and Reformatting of Text

4.5.4 Classification Procedures

4.2 Inter-Machine Links

4.2.1 The Illiac III/PDP-8 Interface

The PDP-8 is to be used as an exerciser for various pieces of Illiac III hardware. To support this adventure the software group is engaged in the following areas:

1. Becoming familiar with the PDP-8 facilities.
2. Attempting to discover the operating procedure and capabilities of certain Dectape system routines with an eye toward possible future modifications.
3. The design of engineering test packages to be used to debug the interface.
4. An investigation into the design of a more comprehensive system which would give the engineer the capability of designing, storing and recalling his own set of test routines.

Area 1 was relatively simple. Area 2 has centered in the area of various cunning bootstrap routines and several of the I/O routines. This apparent simple task has been hampered by various inadequate and often times incorrect documentation.

An initial set of test routines has been written and partially debugged, and will be ready prior to completion of the hardware interface.

To achieve our goal in area 4 means modifying some of the Dectape facilities to allow remote operation of the PDP-8 so that the engineer can remain at the site of the hardware to be tested.

R. Lansford
J. Wick
R. Borovec

4.3 Operating System

4.3.1 Job Control Language

A job control language has been defined consisting of approximately twenty directives covering the following areas:

A. Initialization and Access Privileges

B. File Manipulation

C. Program Assembly and Execution

D. Debugging

Particular attention has been paid to the File Manipulation area. Since our users will initially be using private storage media and eventually be using public storage for their files, the Job Control Language has been designed to bridge this transition smoothly.

L. Katoh

4.3.2 Telecommunications Processing Package (TP^2)

Documentation in this area will be completed in the following quarter. Definition of microtape control is not yet completed.

L. Dunn

4.3.3 Segment Linkage

Work continues on devising a scheme to access multiple segments without the assistance of associative registers. This job is complicated in that one must allow for the possibility of multiprocessing as well as time sharing.

At present several tentative schemes are being evaluated, which all appear to do the job, but with varying degrees of supervisory overhead. Further detailed examination is continuing.

R. Lansford

4.4 Translators

4.4.1 IBAL (Assembler Language)

Work was resumed on Pass 1 of the IBAL translator, written in PL/I for the IBM 360. After alterations were made to incorporate changes that have been made in both IBAL (as specified in the IBAL Manual) and the 360 PL/I compiler, work was centered on the handling of declarations during Pass 1.

A state-transition matrix similar to that used in the rest of Pass 1 is used to process declarations. A structure table is set up for storage of declared structures (including single identifiers and arrays), with attributes and adjustment of node levels in the table according to declared levels and number of subscripts at higher levels.

Reserved words are recognized and those which are storage attributes are indicated in the appropriate places in the structure table. Cell sizes (B, W, H, D) are distinguished from node names by context. Terminal node attributes are evaluated (except for values and lengths of constants) in the structure table in terms of number of bytes or a pointer to a list to handle the LIKE attribute.

Unlike the rest of Pass 1, the declaration section does limited diagnostics in the form of messages regarding syntax errors. For the time being, defaults not specified in the IBAL Manual for certain types of syntax errors are taken so that translation may go on.

Several areas are not yet handled. These include implementation of conditional format items and alternate access structures, and a satisfactory means of incorporating the structure table, subscript bounds, and constants into the block structure of the IBAL program and into the output string produced by the rest of Pass 1.

S. T. Baird

4.4.2 PL/1

Progress under this heading consists of the following document: "PL/1 Declaration Pass: I. Table Structure" by John T. Engle, File No. 771, June 28, 1968

R. Lansford

4.5 Experimental Recognition Procedures

4.5.1 Recognition of Graph Transformation Grammars

Work continued this quarter on the characterization of binary relations, and particularly those binary relations which arise naturally in image description. The goal of this research is to characterize image metastructure so that graph transformations may be made on the basis of a small set of relational properties.

Two file notes were issued. The first, "Properties of a Discrete Space Preserved by Image Processing Relations" by John C. Schwebel and Bruce H. McCormick, File No. 769, July 1968, extends the nine basic properties of relations given in "Consistent Formal Properties of Binary Relations" by Bruce H. McCormick and John C. Schwebel, File No. 762, June 1968, by classifying relations in terms of whether or not they preserve seven basic set-theoretic relations. All sixteen properties are then used to characterize common binary relations of image description.

The second, "Use of Graph Transformations to Characterize An Image: An Illustrative Example" by John C. Schwebel, File No. 770, July 1968, shows how the properties of relations might be used to guide the application of transformations to an abstract graph of an image.

This approach has illuminated some basic properties of the relational structure of images and although it is not clear how extensively these results may be applied, they do appear to be fundamental to further study of image metastructure.

Further work has been done in classifying relations in terms of their relative "precedence" or combining strength with respect to some specific graph transformations. The intersection of these classifying partitions seems to give groups whose members are strongly semantically interrelated. It may be possible to derive these precedence

relationships from our original sixteen properties or from an extended set of properties.

Work is continuing to summarize and unify these results.

J. C. Schwebel

4.5.2 Recognition of Context Free Grammars

During the shortened contract period to August 15, programming of Part I of the scheme for recognition of context free grammars was completed, and testing of this part of the system was begun on a series of sample sets of productions of increasing complexity. Programming was begun on Part II which will "determine" the set of ambiguous productions by building up context elements, using the output from Part I. The output of Part I consists of:

1. The symbol table — in order of appearance of the symbols in the extended BNF productions.
2. Tables of the terminal, non-terminal and anti-terminal symbols, each in alphabetical order, and each determined by the input set of productions, as well as any indication of how many created non-terminal symbols were required.
3. Identification of the unique "distinguished" element if one exists, or else a notation that a unique element does not exist.
4. A listing of the simple Chomsky productions derived from the input of extended BNF productions, listed in groups of productions of length one, productions of length two, and the ambiguous productions.

A. W. McInnes

4.5.3 Recognition and Reformatting of Text

A preliminary list of serials data elements was compiled with a view to the application of LEFT to automatic indexing.

The Rogers-Barnes program for taxonomic classification procedures was examined but has not yet been applied to the matrix of journals versus subject areas.

The report on LEFT language was completed and will be issued shortly.

S. J. Flowerdew

4.5.4 Classification Procedures

A first pass bibliography on classification has been prepared and sent to outside specialists working in this very diffuse area. Additional candidate references have been received from these interrogations and we expect to use these to expand the present bibliography.

J. C. Schwebel

4.6 Design/Fabrication of the Computer

The section numbers of the outline below correspond to section headings of this Progress Report.

4.7 Central System

4.7.1 Taxicrinic Processors

4.7.1.1 Documentation

4.7.1.2 Logical Design

4.7.1.3 Hardware and Wiring

4.7.2 Fast Core Storage Modules

4.7.3 Arithmetic Units

4.7.3.1 Logical Design

4.7.3.2 Implementation

4.7.3.3 Publication

4.7.4 Interrupt Unit

4.7.5 Pattern Articulation Unit

4.7.5.1 Logical Design

4.7.5.2 Documentation

4.7.5.3 Hardware and Wiring

4.7.6 Exchange Net

4.7.7 Status of the Mainframe Assembly

4.8 I/O System

4.8.1 I/O Processor

4.8.2 Channel Interface Unit

4.9 Peripheral System

4.9.1 Secondary Storage System

4.9.2 Scan/Display System

4.9.2.1 Scanner-Monitor-Video Controller

4.9.2.2 Scanner/Monitors

4.9.3 Intermachine Link to Illinet (IBM 360)

4.9.4 Low Speed Terminal Network

4.9.4.1 Low Speed Communications Net

4.9.4.2 Low Speed Buffer

4.9.4.2.1 Low Speed Buffer Control

4.9.4.2.2 Buffer Memory

4.9.4.3 Low Speed Terminal

4.9.4.3.1 Monitor Selectric Typewriters

4.9.4.3.2 Monitor Magnetic Tape Modules

4.9.4.3.3 Teletype Sets

4.9.4.3.4 Analog Instruments

4.10 Power Distribution

4.10.1 Primary D.C. Power Supplies

4.10.2 Power Distribution System

4.10.2.1 Primary D.C. Distribution Center (Room 223)

4.10.3 Control of Power Distribution System

4.10.4 A.C. Power Distribution System

4.11 Unassigned Equipment Pool

4.11.1 Circuit Card Inventory

4.11.2 Test Equipment Additions: Commercial

4.11.3 Test Equipment Additions: Custom-Design

4.12 Documentation

4.12.1 Engineering Manual

4.12.2 Circuit Book

4.12.3 Logic Book

4.12.4 Wiring Table Documentation

4.12.5 Documentation of Opto/Mechanical Design

4.13 Circuit Research and Development

4.13.1 Analog-to-Digital Converter

4.13.2 Control Point Circuit

4.7 Central System

4.7.1 Taxicrinic Processors

4.7.1.1 Documentation

The first version of Volume 1 of the TP manual is currently being edited and prepared for publication. This volume contains the descriptions of the basic registers and local control logic blocks for the TP. These include the base registers, pointer registers, operand stack, the 32 bit adder, and the permuter. Also described are the local control blocks of the above subassemblies, and the control logic for the shift/boolean and algebraic/logical compare blocks.

The second volume of the TP manual, which contains the global control sequencing, is continuing to develop. At the present time the control point flow charts have been written for most of the Primitive Instructions. These flow charts have been developed to give a detailed description of the control signals which must be turned on by each control point as well as the decision logic between control points. Thus the work to convert the flow charts into final logic is minimized.

The signal name directory has continued to expand. As new control logic is developed, the new signal names have been added. In the cases where names have been changed, the old entry is made to indicate that it is obsolete and to refer to the new name. This enables correlation between manuscripts and logic diagrams of varying epochs.

4.7.1.2 Logical Design

The design and implementation of the control sequencing is continuing. A self-consistent philosophy of design has been developed with reference to the use of control points.

4.7.1.3 Hardware and Wiring

The reorganization of the lower bays of the four Taxicrinic Processors was completed during this past quarter. The input cards for the POOL wiring list generating program have been written and are currently being punched and run on the IBM 360.

B. Nordmann

4.7.2 Fast Core Storage Modules

The FABRI-TEK representative, Russell Nelson, arrived July 9, 1968, and resolved most of the initial checkout problems associated with our two core memory systems. Mr. Nelson has agreed to return if any problems associated with "online" operation should arise.

This quarter we received the operating manuals, parts list, and a spare X-matrix board for our two FABRI-TEK core memories.

S. Paul Krabbe

4.7.3 Arithmetic Units

4.7.3.1 Logical Design

The detailed logic design of all registers, data transfer paths and major processing blocks of the AU structure, as distinguished from control, is now complete.

During the last quarter a control strategy was selected. The technique is pseudo-asynchronous = "pseudo" in the sense that true reply signals are not generated by the actual completion of the task, but rather by a timing model (essentially a delay network). The basic sequencing element is the control point, which when activated, enables the appropriate tasks and after some preselected time, resets and enables the next control point. Conditional logic may be inserted, between control points to permit branching and by-passing.

The logical design of the control point has been completed and will be implemented with Texas Instruments Series 7400, I.C. Logic.

Control design using this control point has begun. The control sequences have previously been specified in the AU Simulation (AUSIM) programs and in supporting flowcharts. The flowcharts drawn

for AUSIM use the standard two-way decision boxes. It has been found that a flowcharting convention using an N-way branch notation is more easily identified with the actual control logic. These new flowcharts are being prepared concurrent with actual logic design.

To allow greater flexibility in the use of the I.C. logic, a printed circuit card has been designed with provision for 20 dual-inline I.C. sockets. Wire-wrap connections will be made between the pins of these connectors. As a pilot study in the use of this card, the Exponent Arithmetic Unit was implemented with I.C. logic and found to require only two of these connector cards. Connections between cards must be made via the 44 edge-pins of the card. This pin limitation proved to be no problem in the EAU layout.

4.7.3.2 Implementation

Wiring tables for the 475 cards of the processing hardware (not control) is now 90% completed. The POOL wiring table generation programs developed by P. Krabbe and R. Thomson have proved to be very valuable in this work. The wiring of repetitive sections of logic may be specified with a few "short-hand" statements which in turn may generate hundreds of connections on the final wiring tables.

4.7.3.3 Publication

An article describing the theoretical aspects of the division algorithm employed in Illiac III will appear in the October 1968 issue of the IEEE Transactions on Computers. It is entitled "Higher Radix Division Using Estimates of the Divisor and Partial Remainders".

D. E. Atkins

4.7.4 Interrupt Unit

Design of the register and control of the Interrupt Unit have been rescheduled, and will not be attacked until March 1, 1969.

4.7.5 Pattern Articulation Unit

4.7.5.1 Logical Design

Work has continued on the microprogrammed control of the Pattern Articulation Unit. The branching scheme finally decided upon using three bits to indicate the branching conditions and allow easy access to subroutines. It was decided that recursive subroutine calls would not be allowed because they are not often used and because this restriction simplified the hardware. A paper describing the results of the preliminary investigation is currently being written and will be presented at the ACM Workshop on Microprogramming in Bedford, Massachusetts on October 8, 1968.

The logic drawings for BOOLE control are being updated as a result of errors discovered in simulation.

A slight change has been made in the Transfer Memory. The long-word addresses now range from 8 to 55 instead of 0 to 47. This simplifies some control sequences and insures compatibility with the PAU control.

4.7.5.2 Documentation

Two documents were released during the last quarter. They are Report No. 285 on simulation and File No. 776 describing the display system. See bibliography for complete details. Scheduled for completion during the next quarter are documents on the preliminary microprogrammed control investigations, the implementation of BOOLE control, operation of the stalactite card, and detailed Exchange Net/PAU interface requirements.

4.7.5.3 Hardware and Wiring

An error was discovered in the decoding scheme used on the Local Control Card in the display system. The error was corrected and a 285B etch was approved. The display system design has

been completed and wiring is currently in progress.

The wiring of the Iterative Array is now 3/4 completed.

Both the power and logic wiring of the Transfer Memory are complete. About half of the core planes are required. The TM still requires all power buss wiring, door-to-door wiring and cabling to the Iterative Array to be completed.

R. T. Borovec

4.7.6 Exchange Net

Drafting has completed the drawings for the exchange net to the extent that wiring lists can be made. The wiring list for the exchange net is now 80% complete. It is anticipated that the wiring lists for the exchange net will be complete by November 1, 1968.

Wiring for the SPEU (Simulator for Processor, Exchange or Unit) has been completed. Checkout of SPEU is completed to the extent that we can operate with a fast core module (control byte only). Data transfers will be attempted as soon as the fabrication of interconnecting cables is complete.

4.7.7 Status of the Mainframe Assembly

The following list of items describe mainframe progress during the last quarter.

1. Logical and power wiring completed on Sections 2 and 4 (Transfer Memory). These two sections are now installed in the mainframe.

2. Logical and power wiring completed on Sections 1-upper and 3-upper (Iterative Array). These two half-sections are now installed in the mainframe.

A program has been written and checked out, in PL/1 language, to compile wiring lists. The program is called POOL 1 (Propagator Of Ordered Lists Version 1). This program 1) allows the

user to create a new list, 2) edit an existing list, 3) generates wiring instructions for the user, 4) processes (sort/merge) instructions and forms an ordered list, 5) prints user syntax errors, 6) prints copies of the ordered list. A user handbook is being prepared for this program.

POOL 1 is currently being used to prepare wiring lists for the Illiac III System.

P. Krabbe

4.8 I/O System

4.8.1 I/O Processor

Further design work on the IOP has been stalled awaiting resolution of inter Exchange Net interface communication conventions. However the Exchange Net logical design is now complete and this work can proceed.

Large hunks of the IOP design will be lifted from the TP design now nearing completion.

4.8.2 Channel Interface Unit

An engineering manual of the Channel Interface Unit is in final typing and should be available in the next quarter.

Present scheduling calls for completion of logical design cleanup by mid-December, with wiring lists prepared by mid-January 1969.

B. McCormick

4.9 Peripheral System

4.9.1 Secondary Storage System

No developments are anticipated here until after October 1969 due to budgetary constraints.

4.9.2 Scan/Display System

4.9.2.1 Scanner-Monitor-Video Controller

With the integration of the control flow charts some additional modifications have been made to the basic logic data flow paths in order to simplify the control and in order to unify the operational concepts.

A complete signal name directory for the basic logic was compiled. All of the logical equations for the counter operation and the counter control were formalized.

A paper entitled "Scan-Display System of the Illiac III Computer" was submitted for presentation at the Spring Joint Computer Conference in Boston in May 1969. This paper presents the general ground rules which must be adhered to in order to design a general graphical system, based upon the design experience in constructing one for Illiac III.

V. Tareski

4.9.2.2 Scanner/Monitors

Logic design has been completed and wiring is in progress on a simplified scanner which will permit testing of many electrical and optical parameters of a scanner. The system is the simplest logical assembly which will permit scanning of a selected portion of the total field, gray scale encoding and storage of the scanned information in core. After the scanning operation the contents of the core memory can be repeatedly read to permit display of the information on a monitor.

Techniques have been developed for accurately determining CRT beam current, always a problem for currents less than one micro-ampere. This will allow the best signal to noise ratio consistent with safe operation of the CRT. Beam current is controlled digitally (eight bits) over a range from 10pA to 1μA.

J. Divilbiss

4.9.3 Intermachine Link to Illinet (IBM 360)

During the last quarter an intermachine link was proposed and designed between the Illiac III area and the PDP-8/338 in the Department of Computer Science.

The link provides two 48 bit data registers at the Illiac III end (one for receiving data, one for transmitting) and should prove useful in the area of hardware simulation. Four PDP-8 words are assembled to form one 48 bit word in approximately 15 microseconds.

Included in the link is a teletype which may be used as a remote console typewriter to alleviate the need of running between the first and second floors of the department.

Among the first anticipated uses of the link are exercising the Iterative Array and exercising the Videograph page printer, particularly the character generator.

R. Borovec

4.9.4 Low Speed Terminal Network

4.9.4.1 Low Speed Communications Net

4.9.4.2 Low Speed Buffer

4.9.4.2.1 Low Speed Buffer Control

Due to changes in operating specifications, there are some slight modifications in design now being made. Wiring lists and card allocation will be started on this unit during the next quarter.

4.9.4.2.2 Buffer Memory

Continuing refinements are being made to the tester and this unit should be completed in the near future.

4.9.4.3 Low Speed Terminal

4.9.4.3.1 Monitor Selectric Typewriters

There has been no changes in the status of these units during the past quarter.

4.9.4.3.2 Monitor Magnetic Tape Modules

The type of format and associated hardware required for these units is still being investigated.

4.9.4.3.3 Teletype Sets

(1) Mod 33 Teletype unit has been interfaced with a PDP-8 to provide I/O with Illiac III

4.9.4.3.4 Analog Instruments

No further design has been completed in this area due to prior design commitments.

R. Martin

4.10 Power Distribution

4.10.1 Primary D.C. Power Supplies

All fire damaged power supplies have been repaired, tested, and are now operational.

While it is evident that additional primary supplies will be needed as future current requirements are realized, initial sections of the machine will be adequately supplied by the units now on hand.

All primary distribution cables have been laid and connected to the distribution centers.

4.10.2 Power Distribution System

4.10.2.1 Primary D.C. Distribution Center (Room 223)

The connectors used in the signal and distribution racks have been installed and framework and panels for section 93 have been mounted to the wall.

4.10.3 Control of Power Distribution System

The printed circuit card racks are being assembled and installed into sections 91 and 95. Wiring lists have been initiated and the p.c. card rack wiring should start by the last quarter of this year.

4.10.4 A.C. Power Distribution System

There have been no further additions or changes in this section during the past quarter.

R. Martin

4.11 Unassigned Equipment Pool

4.11.1 Circuit Card Inventory

Because of the large number and types of printed circuit cards being used in Illiac III, a new system for card inventory has been initiated. Since the design of the majority of the sections are complete and the actual required card counts are now known, the printed circuit cards will be allocated and stored per section of the machine. Using this procedure, there should be no confusion as to the location and stock disposition of a particular type of card required for a given section of the machine.

At present, there are about 6500 conventional cards required to complete a minimum machine. These cards are either in stock or have been ordered.

4.11.2 Test Equipment Additions: Commercial

During this period a Dana Model 5500 precision digital voltmeter has been acquired.

4.11.3 Test Equipment Additions: Custom-Design

No additonal custom-designed equipment obtained during this period.

R. Martin

4.12 Documentation

4.12.1 Engineering Manual

The Engineering Manual has been updated with the latest circuit and logic drawings.

4.12.2 Circuit Book

Twenty nine circuits or variations of circuits were released for distribution to the Circuit Book. Six additional circuits are under documentation at this time.

4.12.3 Logic Book

General revision of the logic documentation for the taxicrinic processor were undertaken. Revisions were of corrective and design nature. The revision of these drawings will be complete by mid-October.

Interim logic documentation for the transfer memory unit are near completion. Approximately sixty drawings were completed during this quarter.

Exchange unit logic documentation has progressed very near completion. A total of thirty four drawings were completed.

All other logic documentation has not significantly changed status since last report.

4.12.4 Wiring Table Documentation

Wiring tables for transfer memory, iterative array, iterative array supplement, two I.C. cards for the Arithmetic Unit, and wiring list for SPEU simulator (for processors, exchange or unit) are completed.

Units in partial completion are Taxicrinic Processor 90%, Arithmetic Unit 80%, Exchange Net 80%. It is anticipated that all three of these units will have completed lists by November 1968. Completed lists exists as POOL I except the Iterative Array which is handwritten.

J. Otten

4.12.5 Documentation of Opto/Mechanical Design

From July 1, 1968, through September 30, 1968, nine (9) new detail and layout drawings have been made.

Seven (7) additonal drawings have been processed according to engineering change orders.

Three (3) work orders have been placed with our machine shop.

S. Zundo

4.13 Circuit Research and Development

4.13.1 Analog-to-Digital Converter

A greatly improved high speed digital-to-analog converter system has been developed and packaged in printed circuit form. This new system, employing several important innovations, will be fully described in a forthcoming file note.

Circuits for the generation of functions sine, cosine and logarithmic have been developed, tested and (for the trigonometric functions) reduced to printed circuit form.

4.13.2 Control Point Circuit

An IC control point circuit satisfactory to most, if not all, of our logic designers has been designed. A very important part of this was development of a modeling (i.e., timing) circuit which would be easily adjustable, stable, noise insensitive, repetition rate insensitive, highly assymmetric (i.e., different delays for the two transition directions) and presenting no hazard to the safe operation of the IC's.

J. Divilbiss

4.14 Bibliography

During the past quarter the following reports and file numbers were issued:

Reports

Report No. 275	Borovec, Richard T., "The Logical Design of a Class of Limited Carry-Borrow Propagation Adders", August 1, 1968.
Report No. 285	Koo, Ping L., Borovec, Richard T., "The Pattern Articulation Unit of Illiac III Simulation: Part I—Iterative Array Transfer Memory——BOOLE Control", September 20, 1968.
Report No. 290	Koo, Ping L., Atkins, Daniel E., "Arithmetic Unit of Illiac III: Simulation and Logical Design—Part II", October 28, 1968.

File Numbers

File No. 769	Schwebel, John C., McCormick, Bruce H., "Properties of a Discrete Space Preserved by Image Processing Relations", July 18, 1968.
File No. 770	Schwebel, John C., "Use of Graph Transformations to Characterize an Image: An Illustrative Example", July 18, 1968.
File No. 771	Engle, John T., "PL/1 Declaration Pass: I. The Table Structure", June 28, 1968.
File No. 773	Krabbe, S. Paul, "Specifications for a 9" False Floor for DCL 223 That Overlaps Existing False Floor in DCL 280", August 7, 1968.
File No. 776	Borovec, Richard T., "The Pattern Articulation Unit of Illiac III Display System: Design and Implementation", September 20, 1968.
File No. 550-116	Krabbe, S. Paul, "Supplemental Specifications for Wire-Wrap Wiring of Sections 1 and 3 (PAU) of the Illiac III Computer System", August 13, 1968.
File No. 550-117	Serio, F.P., Pelg, E., "Procurement and Design Specifications for Printed Wiring Boards", September 23, 1968.

Outside Talks:

McCormick, Bruce H., "Graph Grammars for Image Processing", Joint meeting with the Society of Photo-Optical Instrumentation Engineers and the Pattern Recognition Society, Washington, D.C., August 25, 1968.

McCormick, Bruce H., "Seeing and Believing by Machines", Sixth Annual Allerton Conference on Circuit and System Theory, Allerton House, Monticello, Illinois, October 2, 1968.

Borovec, Richard T., "A Report on the Use of Microprogramming for the Illiac III Image Processor", ACM Workshop on Microprogramming, Bedford, Massachusetts, October 6-9, 1968.

4.15 Illiac III Staff[1]

Senior Staff

Dr. James L. Divilbiss - Principal Research Engineer
Dr. Robert M. Lansford - Research Associate
Professor Bruce H. McCormick - Principal Investigator
Professor Sylvian R. Ray
Professor K. C. Smith - Consultant

Professional Staff

Robert C. Amendola
Richard T. Borovec
Lawrence A. Dunn
Mrs. Linda M. Katoh
Ronald G. Martin

Electronic Engineering Assistants

David L. Foster
Thomas E. Kingery
S. Paul Krabbe
Edmund Pelg
Frank P. Serio
Joseph V. Wenta

Electronic Technician II

John L. Fileccia
Robert G. Thomson

Electronic Technician I

Eugene E. Althaus
George T. Lewis
William A. Miller
Bennie D. Williamson

Senior Laboratory Mechanic

Ralph M. Rivera

Engineers

*Daniel E. Atkins
*Stuart Baird
*Richard T. Borovec
*Thomas T. Chen
*Stanley J. Flowerdew
*Sergio Franco
*Lakshmi Goyal
*Richard P. Harms
*Bernard J. Nordmann
*Val G. Tareski

Programmers

*James P. Fornango
*Allan W. McInnes
*John C. Schwebel

Drafting

John Otten
Stanislavs Zundo

Secretarial

Mrs. Donna J. Stutz

[1] Not including hourly employees
*Research Assistants

5. ILLIAC IV

This work was supported in part by the Department of Computer Science, University of Illinois, Urbana, Illinois, and in part by the Advanced Research Projects Agency as administered by the Rome Air Development Center, under Contract No. US AF 30(602)4144.

REPORT SUMMARY

The ILLIAC IV Advisory Committee Meeting was held at Burroughs Corporation, Paoli, Pennsylvania, on July 17 and 18. Personnel from Rome Air Development Center (RADC), Advanced Research Projects Agency (ARPA), Burroughs, Texas Instruments (TI), and the University of Illinois attended the meeting. Also present were the members of the ILLIAC IV Advisory Board and of the Advisory Committee to the National Academy of Sciences on the NIKE-X System.

Technical presentations were made on the system design, the programming activity, the diagnostic system, and the application of ILLIAC IV with phased array radar. Laboratory demonstrations of the breadboard PE and of the thin film memory system were made. The concensus of opinion was that the meetings were excellent and presented the attendees with a detailed report of the ILLIAC IV Project.

The B5500 installation continues to operate well. The use of a dedicated machine has had tremendous impact on the University's effort to develop software. The software effort would have been hindered if the machine were not available on a dedicated basis. Additional magnetic tape drives and disk modules have been installed. Moreover, the predicted load from the design automation and diagnostic effort has resulted in the University ordering an additional processor and I/O equipment.

The data link between Burroughs and the University has been installed, and the Mohawk data terminals provide tape to tape transmission. Design Automation data which is to be run on the B5500 has been transmitted to the University by the data link.

The major diagnostic effort is the checkout of the breadboard PE. Test programs to run on the PE exerciser have been provided by the University. The checkout efforts have been going slower than anticipated, and efforts have been initiated to speed up the checkout of the functional logic.

In software, several areas showed progress. In the ISL translator, improvement of its speed and several program sophistications were made. Changes were made in the syntax preprocessor to increase its efficiency, and the complete documentation of the syntax preprocessor was begun. The CAT language project was divided into four areas to increase its progress and development. The supplementation of Glypnir Version I was, with the exception of procedures, completed. A Version I user's manual is in preparation. Progress was also shown in the syntax and semantic description of SQUASH, the debugging aid to ALGOL.

The ILLIAC IV education consisted of organizing and presenting a course of instruction for ILLIAC IV personnel. The subjects discussed by this course were: ALGOL, the B5500, the ILLIAC IV assembler, Tranquil, and ILLIAC IV applications.

Mr. Richard Stokes of Burroughs has been replaced as Deputy Project Manager by Mr. Walter Fresch. The University has approved the replacement.

The major problems are: Debugging of the PE breadboard and obtaining artwork for the PE and CU printed circuit boards. As mentioned previously, the University will cooperate with Burroughs in the PE debugging by use of the PE logic simulator. The artwork production is being followed very closely, and weekly reports are being received from Burroughs. The University's B5500 facility is available for the Design Automation Activity from 12:00 midnight to 8:00 A.M. nightly, and additional time is provided on weekends. The availability of the computer time and quick turn-around will help the DA effort.

The University has requested Burroughs to use the University's computer to reduce the cost of running the DA programs. The funds from ARPA will carry the project to mid-March.

5.1 Diagnostics

5.1.1 PE Logic Simulator

5.1.1.1 Generation of PE Logic Simulator

The PE Logic Simulator (or the Test Simulator Program) was improved and completed this quarter. The simulator takes two-ten seconds of the B5500 processor time for one clock of the PE.

The simulator body is, basically, a set of ALGOL procedure statements whose identifiers correspond to package types while the actual parameters correspond to the signals incident to the package. There is one procedure statement for each package, and the order of the statements is defined by the level assignment of the packages.

Because there are several looped packages, it is impossible to follow the logic in one sequence; therefore, there is a program loop for each set of the looped packages for evaluating the logic equations until all outputs of the packages are stabilized. If, after thirty times of repetition for the looped packages, the outputs of the packages are not stabilized, a message will be printed out to indicate a possible race condition.

A few problems were caused by the usage of a compiler language as a simulator media. Since the number of words in a program segment cannot exceed 1023, some redundant words as BEGIN, FORMAT, and END were inserted to divide the segment. About four minutes of the B5500 processor time was used to generate the simulator body, which is in a file containing approximately 4000 card images.

An editing program combines the generated simulator body with the procedure declarations and the input/output processing program. The procedure declarations describe the logic of all the package types in ALGOL. The formal parameters of the procedure correspond to sixteen or eighty pins of the package.

The input to the simulator may contain specifications of a microsequence and/or data to be put into the PE and special symbols to control the content of the output from the simulator. The content of every register and the signal values of some combinational circuits, as well as an array showing the state of all the inter-package signals for the convenience of logic debugging, can be printed out by the use of control symbols.

5.1.1.2 Level Assignment and Loop Detection

Before the generation of the simulator body, a level had to be assigned to each package. This was done by referring to the arc list which is a reduced form of the wire list. Since the level is not assigned to a package by the logical equations but by the arc list, some packages constitute a loop; therefore, to assign a meaningful label to all packages, the steps listed below should be followed.

I. Assign the level to the packages in the ascending order and in the descending order. Extract the packages whose levels are not defined -- these packages may constitute loops.

II. Distinguish each loop from the set of extracted packages. Looped packages are reduced to one pseudo-package.

III. Assign the level to the set of pseudo-packages, and after assigning the level, expand each pseudo-package to the original packages.

All packages are converted to the compact symbolic names within these programs to increase the speed of level assignment. There are several programs for this conversion.

The simulator can treat up to 2000 arcs and 500 packages with up to 1000 arcs involved in loops. There are about 1800 arcs and about 450 packages (including some dummy packages) in the PE. The number of arcs within the looped packages is 156 and the number of packages is 482. The processor time for these programs is four minutes for the first step, thirty minutes for loop detection, and four minutes for the last step.

5.1.1.3 Application to Logic Debugging

The simulator has been used to debug the logic design and the wiring list of the PE. The basic transmit and arithmetic instructions were tested on the simulator, and Burroughs was informed of the detected errors.

5.1.2 Generation of PE Diagnostic Programs

5.1.2.1 Path Tests

The last two subprograms of the Test Ordering Program (TØP) were written and debugged. The algorithm used in the ordering of test cases is based on the evaluation of a weight for each test. It has a tendency to prolong the chains of "success" branches compared with those of "failure" branches in the test tree. This algorithm is expected to be more powerful than other methods in locating multiple errors.

The following are major files generated by the TØP.

I TØP/NBRTWIG: It contains an integer which indicates the size of the following three files. At present this integer is 831.

II TØP/PTHNAME: This file specifies the path to be tested or the failure location (at the termination). This file is a one-dimensional array.

III TØP/SUCBRNH: It is a one-dimensional array. The value tells where to branch if it is non-zero. Zero indicates a termination of the tests.

IV TØP/NODELBL: This file is also a one-dimensional array, and it indicates the branched-to location. Some tests are labeled.

The path test sequence can be specified by the previous four files. Some additional information about failure locations (e.g., equivalent failure location and set of equivalent multiple failure locations) should be referred to when the path test sequence is used. An additional program is being written to combine these

files into one file having the format of the PE Exerciser Test Assembly language.

The output of the Path Test Generator has been reviewed In the input and in the program, a few errors were removed.

5.1.2.2 Combinational Tests

Some final additions and modifications were made to the Combinational Test Generator during this quarter. The expected response procedure for BSW was added, and the procedure for CSA was modified slightly to perform the expected response calculations for MSG and MDG tests. All expected response procedures are now operating, and few additional modifications are anticipated. The documentation for CTG was also completed during this quarter.

Several programs were written which translate the output of CTG (in PEX assembly language) into a form suitable for input to the PE simulator. An old version of the PEX Input Generator (PIG) was rewritten to facilitate the use with the combinational test record. This program translates control signal names into a fixed format for the PEX Test Assembly Program (PEXTAP). Another program, Assembled Code Translator, written during this quarter is used to convert the machine code output of PEXTAP into the form required by the simulator.

With these three programs (PIG, PEXTAP, and ACT), any program written in the PEX assembly language can be run on the PE simulator. Running of sample combinational test programs provides a means for checking the programs themselves, for checking the expected response calculations in CTG, and for possibly detecting design errors in hardware.

A large sample of the CPA and ADA tests was run on the simulator. These runs indicated that the CPA tests, CPA expected response calculations in CTG, the simulator, and the PE were all functioning properly. In the case of ADA, the expected response calculation in CTG and the polarity of one signal in the PE were found to be in error. With these errors corrected, the address

adder tests, CTG, simulator, and PE were also in agreement for ADA. Presently, a sample of the Barrel Switch tests is being run. It is hoped that future runs of these programs will aid Burroughs in its debugging efforts and will serve to eliminate any errors in the combinational test programs and the expected response calculations.

5.2 Design Automation

Final specifications for the Delay Check Program were completed in the first part of this quarter. The program was written and tested on small test boards here at the University. Final debugging and testing are held up until the Post-Processor Program is operational. The work to make this program operational will continue into the next quarter.

Arrangements which will enable Burroughs DAS production runs to be done on the B5500 here at the University are nearly complete. A small software package has been written to convert program data files into tapes for the Mohawk Data Set and for their reconversion. This operation should increase the use of the B5500 here at the University and give the Burroughs group added machine capability.

Work has begun on a preprocessor program for an ILLIAC IV Design Automation System. This program will input the data to the program, convert it into a suitable data file, and check for user syntax errors. The program will also prepare useful reports for aiding a designer in verifying his program input, logic diagrams, and equations.

5.3 Translator Writing System and Language Development

5.3.1 Introduction

Progress was shown in many areas during this quarter. Some of the progress involved changes which increased the efficiency in programs and the development of a new language. Improvements were made in the parser, and there was also progress in converting the parser instruction table to an ALGOL program. The ISL translator, during this quarter, was improved; and the Pass II of the Tranquil compiler was begun. The following paragraphs discuss these and other areas of progress.

5.3.2 Syntax Preprocessor

The implementation of minor changes in the syntax preprocessor continued. The purpose of these changes is to increase the preprocessor's efficiency. A line by line rewriting of the code which is also taking place will increase efficiency.

The inception of complete documentation of the syntax preprocessor has begun. The initial emphasis of this documentation is the attainment of a detailed description of the algorithm used in developing the syntax preprocessor.

5.3.3 Parser

A much more efficient version of the parser was implemented and debugged. All the indirect addressing was replaced by direct transfer addresses. The parser instructions were expanded to six bits allowing new, more efficient parser instructions to be generated and allowing the use of stream procedures for fetching or testing parser instructions and operands. The new parser has run from 100 to 750 cards per minute on test programs of various complexity.

The new parser which implements more meaningful error and monitoring messages has an expanded error recovery scheme which actually corrects errors in certain situations. By having many tests and sorts performed in the syntax preprocessor, the testing done at parse time is kept to a minimum.

Work is nearly finished on a program that will convert the parser instruction table to an ALGOL program. This should add another significant increase to the speed of the parser since most procedure calls and array accesses will be eliminated by this approach.

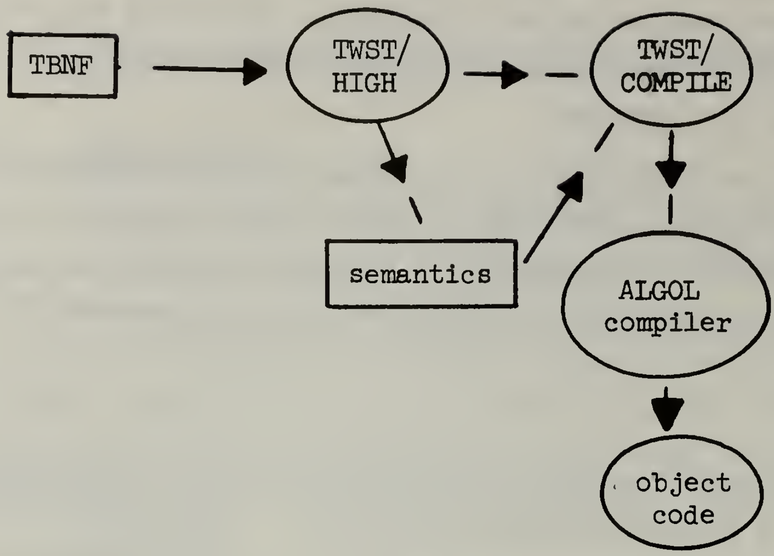
5.3.4 Twinkle

Another area of effort was the description and implementation of Twinkle, a syntax description language. It will not only combine all the features of present syntax languages used by the ILLIAC IV Translator Writing Systems but will also contain additional features. A TWS generated recognizer for this language will become the syntax scanner for the syntax preprocessor.

5.3.5 TWST/TBNF

A new version of TWST which translates from TBNF (translatable BNF, i.e., extended BNF) to Burrough's code has been completed and seems to be reliable. It offers a two-three fold speed increase over the table driven version. A precedence notation compatible with the scheme has been planned but not yet implemented. Tapes and programming manuals for TWST/TBNF are available.

Schematic diagram



5.3.6 TWS Semantics - ISL Translator

During this quarter, a high priority was given to the improvements in the brute-force ISL translator in order to make the task of the high-level language groups easier. The speed improvement effort was successfully completed. Several program sophistications were introduced in the translator which resulted in a speed improvement by a factor of six. The present brute-force translator can now operate, on the average, at 1200 cards per minute of processor time. Also completed was the addition of control cards to the translator.

Work is now continuing in two areas: 1.) The TWS-ISL translator is being completed, and pass-n facilities are being added to both translators; 2.) Being started is a documentation effort whose goal is to produce, as soon as possible, an ISL user's manual containing examples of semantic descriptions of programming languages.

5.4 Tranquil

The Tranquil work for this quarter involved three areas. One of the work areas concerned storage allocation in Tranquil. In the instances that all data fit in PE memory, the compiler accepts them and allocates the spaces for them. The array type data is cut into blocks of size 256 x 256. Operations and I/O between the disk and the memory are done in terms of these blocks.

The compile time allocation of space in the CU was another effort area. Routines were written for the compile time allocation of space in the CU and further work was done on the use of storage schemes for sets. The CU allocation routines involve a priority system and a compile run-time stack and include CAR and LDB usage. The set schemes are linked with their declarations, and the amount of information given or found determines whether dynamic allocation needs to be incorporated.

The third area of Tranquil effort involved the beginning of the major programming effort for Pass II. The overall structure of Pass II was laid out and programmed. Partially completed were programs to analyze assignment statements for their meaning and for compiling. The algorithm for determining non-dynamic variables was implemented.

5.5 Glypnir

Implementation of Glypnir Version I was, with the exception of procedures, completed during the third quarter of 1968. Procedures will be implemented during the fourth quarter. The code generated by the compiler has been debugged syntactically on the current version of the assembler, and complete debugging will begin as soon as the new ILLIAC IV simulator becomes available. A Version I user's manual is in preparation, and copies of the rough draft can be obtained.

5.6 SQUASH

The syntax and semantic description of SQUASH, the debugging aid to ALGOL, has progressed. The syntax is virtually complete and

has been successfully accepted by SYNPROF, the TWS syntax processor. However, minor modifications in syntax may be necessary to facilitate the writing of semantics. But, having considered semantics, coding has begun.

5.7 System K

5.7.1 Introduction

The main activities of the SYSK group this quarter concerned many areas. These areas were the ILLIAC IV assembler, simulator and loader, design of the B6500 operating system, and development of OSK prototype features.

5.7.2 Assembler

A new version of the assembler was written to serve as the foundation of the forthcoming macro assembler. Much time was spent in making this assembler as fast as possible. The assembler is substantially complete. It emits pseudo-orders for the loader; it accepts the latest ILLIAC IV order code; it creates a cross-reference table; and it has extensive file manipulation facilities to aid the programmer in maintaining large programs. Extensions are now being added for TMU (the Test Maintenance Unit on the ILLIAC IV) commands. This assembler with the TMU command mode will be sufficient for the B5500 ILLIAC IV operating system at Paoli, Pennsylvania.

5.7.3 Simulator

The development of an efficient, single quadrant ILLIAC IV simulator was slowed because all the operation code assignments and the instruction semantics were changed. But the simulator coding is now substantially complete. A timing simulator was written and will be integrated into the simulation package. Simulator maintenance will be handled by a new individual so that the presently involved person is free to work full time on the loader. The transition will occur on the first of October.

5.7.4 Loader

The basic functions of the ILLIAC IV loader have been defined. In particular, the loader related code information (address relocatability, storage assignment, external references, etc.) is now fixed, and its fields and values in the object code file assigned. The new assembler emits this loader information, and the new simulator includes a loader to interpret this information.

5.7.5 OSK (ILLIAC IV Operating System) Development

There was considerable discussion of the alternative means of handling IOC interrupts on the B6500 and of the structure of the B6500 control programs. The write-up of the summarization of the decisions is in preparation. During the next quarter, a prototype version will be coded to work with the new simulator.

5.7.6 Interim OSK Features on the B5500

Progress is being made on integrating ILLIAC IV languages with the B5500 system. The deck needed for a simple assemble and execute run is listed below.

```
?USER = LP
?COMPILE  LP/TEST WITH ASK LIBRARY
?DATA CARD

 ____
 ____       ASK assembly program
 ____
 ____

?EXECUTE  LP/TEST
?END
```

Work is continuing on a simulator scheduler for the B5500. The goal of this project is to call the simulator into execution whenever the B5500 is not busy on foreground tasks. No operator will be required because the simulator initiation will be automatic. Also, no user will be able to destroy the queue of simulations since the

scheduler will survive system hang-up, halt loads, and any disaster that does not destroy the disk (so far no user has ever destroyed the disk). Simulations will be scheduled using a prototype user services subsystem. The scheduler will work with any other long-running job that is suitably halt-load proofed.

5.8 CAT

5.8.1 General Compendium

The CAT language project has been divided into four distinct efforts whose results will fit into the ILLIAC IV system in various places. This is a result of meetings that were held here this summer with groups of users. The four effort areas are as follows: 1) general I/O routines which will add to Tranquil some kind of disk read and write statements using symbolic file names; 2) a descriptive geometry language which will allow the user to specify his problem space in a general notation rather than forcing him to tediously define every index set; 3) a study of storage allocation schemes for both fast memory and disk with a view toward implementing general techniques within a compiler (a possible result may be automatic I/O which makes the ILLIAC IV disk appear to be an extension of memory); and 4) a study of the optimization of disk I/O by linear programming techniques which reorder the arithmetic statements within a code to allow efficient use of the data set.

In addition to the usefulness of each of these studies as an individual part of the ILLIAC IV system, it is hoped that they can be integrated into a software package for the general user. In this line, a study is also being made of the non-mathematical parts of several large programs; since such "data processing" seems to account for as much as eighty percent of the total run time in many computing centers, an efficient and general system on ILLIAC IV is evidently highly useful.

5.8.2 General Optimization

In the fourth area mentioned above, work began this quarter. This work is involving an attempt to minimize disk latency for non-core contained ILLIAC IV problems. In so doing, permutation of source code, as well as dynamic relocation of data on the disk, is being considered.

At present, the method of approach involves quantizing the time axis over which any computation will take place. Binary variables are then introduced which have values corresponding to each of the actions taken (e.g., compute, input, overlay, etc.) during the particular time interval. It is hoped that, with only a few reasonable simplifying assumptions, constraints may be placed upon these binary variables. The difficulty is keeping these constraints linear.

5.9 Mathematical Applications

5.9.1 Partial Differential Equations

During this quarter, a research assistant spent six weeks at Los Alamos Scientific Laboratory, Los Alamos, New Mexico. A Tranquil code was written for the Particle in Cell (PIC) method used in hydromagnetic theory. While at Los Alamos, much machine time was used for collecting statistics on various storage schemes for particle in cell methods as applied to a parallel computer. The future plan is to fully document this Tranquil code and the collected statistics during the next quarter.

5.9.2 Ordinary Differential Equations

Work was completed on a system of equations for metabolic systems, and this system was summarized in ILLIAC IV Document 197 [1]. The assembly code for this problem is being kept up to date as newer versions of the assembler are available.

Further study is being done on a system of equations related to the physics of a muonic atom. These equations are:

$$\frac{dF_j^K(r)}{dr} = \frac{K_j F_j^K(r)}{r} + (B + U_0(r) + W_j) \quad G_j^K(r)$$

$$+ U_2(r) \sum_i A_{ji} \; G_i^K(r)$$

$$\frac{dG_j^K(r)}{dr} = \frac{-K_j G_j^K(r)}{r} + (2 - B - U_0(r) - W_j) F_j^K(r)$$

$$+ U_2(r) \sum A_{ji} F_i^K(r)$$

The object is to find an exact value for the eigenvalue B. Given an approximation to B and suitable boundary conditions, integration takes place from the inner and outer boundaries to a central "fitting radius". The match of the results at this radius determines the new value of B for the next iteration of this process.

For the large problems in this area, the number of equations is larger than the number of PE's available, and a method must be devised for sharing the computation of the equations over these PE's. Assembly codes are being written to test different but like strategies and to do the actual computation.

5.9.3 Alternating Direction Iteration Scheme

In these three months, an alternating direction iteration scheme in ASK was written, compiled, and time simulated. The time simulation consists of running the program for one and two iteration cycles on the Sankin Time Simulator and separating the I/O $\sim 1.10 \times 10^{-3}$ secs from actual iteration time $\sim 2.3 \times 10^{-3}$ secs.

The code is now in the final stages of simulation, in comparison to the same code in ALGOL, and in documentation. The solution matrix has a small oscillation in it, otherwise the simulation is complete. Two important facts will be the result of the comparison of this code to an identical ALGOL code. One, it will check our results in computation, and two, it will give a comparison of the time simulation to conventional machines. The written report will follow the completion of these two remaining problems.

Also, a successive, over-relaxation iteration scheme in Tranquil has been compiled. This scheme is presently being rechecked for computational errors.

5.9.4 Hydrodynamic Codes

The numerical solution of the Eulerian hydrodynamic equations, in two-dimensional cartesian coordinates, was coded in Tranquil using checkerboard storage. An additional code which traces the path of

"mass-less" particles through an Eulerian grid was also completed during this quarter. Both codes have been syntactically debugged for the Tranquil compiler. An ILLIAC IV Document which will describe the methods used and the codes will appear soon.

5.9.5 Boltzmann's Equation

The ILLIAC IV Document 200 [2] describing the implementation of a Monte Carlo method for evaluating the Boltzmann collision integral to ILLIAC IV was completed during this quarter. The method that is being used for the evaluation of the Boltzmann collision integral was developed by Arnold Nordsieck and Bruce L. Hicks of the University of Illinois [3]. The implementation of this method for ILLIAC IV requires the generation of random numbers in each PE which are also random with respect to the random numbers being generated by the other PE's. Different random number generators for ILLIAC IV which will insure this requirement are being considered. A Tranquil code will be written for this method of evaluating the Boltzmann equation.

5.9.6 Matrices

An algorithm to find the solution matrix X to a symmetic matrix A by using the square-root method (Cholesky [4]) was coded in Tranquil. The order of the matrix A was $n = 64$, but it can without any difficulty be extended to any magnitude $n \leq 256$.

From theory it is known that a symmetric matrix can be written in the form $A = SS^T$. Using the upper triangle of A, the transpose S^T is then given by

$$s^T_{11} = \sqrt{a_{11}} \quad , \quad s^T_{ij} = \frac{a_{ij}}{s_{11}} \quad j > 1.$$

Further, $s^T_{ii} = \sqrt{(a_{ii} - \sum_{\ell=1}^{i-1} s^2_{\ell i})} \qquad i > 1$

$$s^T_{ij} = \frac{a_{ij} - \sum_{\ell=1}^{i=1} s_{\ell i}\, s_{\ell j}}{s_{ii}} \qquad j > 1 \text{ and } i < j$$

$$s^T_{ij} = 0 \qquad i > j.$$

With $A = SS^T$ and $Ax = b$, the result of substitution is

$$SS^T x = b.$$

From this $S^T x = y$ and $Sy = b$.
To find y from the lower triangle S, the following formulas are used.

$$y_1 = \frac{b_1}{s_{11}} \quad , \quad y_n = (b_n - \sum_{\ell=1}^{n-1} s_{n\ell}\, y_\ell) | \, s_{nn} \quad , \quad n > 1 .$$

Having y, x is found by using the upper triangular matrix S^T and back substituting:

$$x_n = \frac{y_n}{s^T_{nn}} \quad , \quad x_k = (y_k - \sum_{\ell=k+1}^{n} s_{k\ell}\, x_\ell) \, | \, s_{kk} \quad , \quad k < n.$$

The problem of having a negative argument in determining the roots in S^T has been taken care of by using the marker "-" (minus) with the rule in mind that "-" $\times$ "-"="-".

Then s^T_{ii} becomes:

$$s^T_{ii} = - \sqrt{|a_{ii} - \Sigma \; s^2_{\ell i}|} \quad .$$

Considering the formulas above, a "straight" storage scheme for all matrices involved is suggested. To illustrate this

point, let us look at an 8 x 8 matrix S^T. Assuming the elements $s_{ik} \epsilon S^T$ with k=i, i+1, ..., 8 for $i \leq 3$ have been calculated, then the following is the case:

$$
\begin{array}{cccccccc}
s_{11} & s_{12} & s_{13} & s_{14} & s_{15} & s_{16} & s_{17} & s_{18} \\
 & s_{22} & s_{23} & s_{24} & s_{25} & s_{26} & s_{27} & s_{28} \\
 & & s_{33} & s_{34} & s_{35} & s_{36} & s_{37} & s_{38} \\
 & & & s_{44} & s_{45} & & &
\end{array}
$$

With s_{44} already known, the calculation of s_{45} is wanted.

$$s_{45} = [a_{45} - (s_{14}\ s_{15} + s_{24}\ s_{25} + s_{34}\ s_{35})] \mid s_{44}$$

The elements above s_{44} and s_{45} are involved in this calculation, or stated more generally, it is: To calculate s_{ik}, $k > i$, only the elements of columns i and k which are above s_{ii} and s_{ik} enter the calculation. Furthermore, since the program has been written in such a way that the elements in row i, which are to be determined, are under SIM control, the efficiency of a "straight" storage scheme becomes even more apparent.

5.9.7 Eigenvalues

During this quarter, work was done in the investigation of eigenvalue problems. A code in assembly language has been written for Jacobi's Method for finding eigenvalues. The algorithm used in the code is a modification of the classical Jacobi Method. The matrix is divided into 2 x 2 sub-matrices along the diagonal, and successive orthogonal transformations are used to eliminate the off-diagonal elements of each sub-matrix. With each iteration, n elements of an n x n matrix are eliminated. The code is presently being debugged, and various storage schemes are being considered. The code is being

timed both on the SANKIN Simulating System and on the SIM/TIME Simulator. A timing estimate will also be derived from the IBM 360 for purposes of comparison.

5.9.8 Root Finding

During this quarter, work continued on adapting Lehmer's algorithm to a parallel machine. Lehmer's algorithm determines whether or not a polynomial has a root in a given circle. Using Gerschgorin circles, the center and the radius of a circle in which all of the roots of a polynomial are found can be obtained. The problem is how to efficiently cover this area with 256 circles so that the percent of duplication is minimal.

The first approach was to divide the area obtained by Gerschgorin circles into 256 identical squares. This area was then covered by circumscribing circles around the squares. The next time, this area was covered by inscribing circles, and the missed area was covered with other circles. In both of these cases, the overlap was fifty-seven percent.

To reduce this overlap figure, the second approach was to divide the area into 256 identical hexagons. In this case, only circumscribing circles were tried, and the duplication was twenty-one percent. Future work will be directed toward reducing the overlap figure more.

5.9.9 Special Functions Subroutine Library

This quarter, work has continued in ILLIAC IV's special functions subroutine library. All previously coded subroutines have been recoded in the latest version of the ASK assembly language. The question of subroutine linkage was also considered in the rewriting of these codes. It was decided that the function argument should be in the A register upon entering the subroutine, and its evaluation would be left in the A register. It was also decided that the subroutine itself would save all of the other registers, except the B register, and restore these at the end of the routine. The functions

will be evaluated only in the enabled PE's, and all other PE's will be left alone.

New 64-bit codes have been written for natural logarithm and arctangent. Also, a new 64-bit code has been developed for square root which completely eliminates division.

I. Natural logarithm: Let y be the number of which it is desired to find the natural logarithm.

Then

$$y = 2^i m \qquad \frac{1}{2} \leq m < 1$$

where i is the floating point number exponent and m is the floating point number mantissa.

The approximation of $\log_2 m$ is made by a polynomial for $\frac{1}{2} \leq m < 1$. Then the natural logarithm is evaluated in accordance with the relation:

$$\ln y = (i + \log_2 m) \cdot \ln 2$$

II. Arctangent: The approximation of arctan (x) is made by a polynomial for $x \in [0, \operatorname{Tan} \pi/8]$.

For $x \in (\operatorname{Tan} \pi/8 , \operatorname{Tan} 3\pi/8]$

$$\operatorname{Arctan}(x) = \operatorname{Arctan}\left({}^{x-1}/_{x+1}\right) + {}^{\pi}/_{4}$$

For $x \in (\operatorname{Tan} {}^{3\pi}/_{8}, \infty)$

$$\operatorname{Arctan}(x) = \pi/2 - \operatorname{Arctan}({}^{1}/_{x})$$

The above three cases all use the same approximating polynomial.

III. Square Root: Let A be the number of which it is desired to find the square root. The iterative scheme used is:

$$x_{n+1} = x_n/2\,(3 - A\,x_n^{\,2})$$

to approximate $1/\sqrt{A}$

Thus $\sqrt{A} = A \lim_{n\to\infty} x_n$

Starting points for this iteration are approximated by second degree polynomials.

5.9.10 Long Codes

As a first step in this task, preliminary studies were done of the Theory of Stability of Motion and of Canonical Transformations. The behavior of the solutions for equations of motion having the form $\dot{x} = Ax$ was investigated. (It was assumed that A was a constant coefficient matrix.) Also, the criteria of stability of the above-mentioned autonomous systems were reviewed. Finally, a detailed illustrative example was worked out for showing the effect of errors in the observation of the initial vector x_0 on the solution of the equations of motion for a given autonomous mechanical system.

5.10 Linear Programming

During this quarter, specification of the mathematical procedures for the first linear programming system, LPS, has largely been completed. Procedures for handling vector bounds, ranges of the right hand side, and basic parametric programming have been drafted. Also during the quarter, the group has examined applications of linear programming to problems of hardware design, along lines suggested by Dr. Masao Kato during his visit to the University.

The solution of a linear programming problem in standard form is handled by a modification of the revised simplex method, product form. The original matrix is partitioned to produce a degree of parallelism suitable for ILLIAC IV. Rows are assigned to specific PE's in a manner designed to distribute calculations evenly among the PE's. The matrix is skewed within and across quadrants to facilitate vector updating and reduced cost calculations. Multiple pricing has been adopted to minimize the number of iterations and disk accesses, thereby reducing overall calculation time and increasing accuracy. Tests have been initiated to evaluate the efficiency of the algorithm with regard to PE utilization and storage allocation.

It is necessary at certain points in the solution procedure to recalculate the updated inverse in product form, minimizing the number of non-zero elements obtained while maintaining a high degree of accuracy. Work continues in the development of such a reinversion procedure from the several techniques which have been examined.

5.11 Radar Processing Applications

Some of the efforts during these three months have been involved in the conversion of the Kalman Filter tracking programs to 32-bit floating point mode. The development of a Tranquil version of the Kalman Filter and the analysis of the NISIM (NIKE Simulation) programs from Bell Telephone Labs were other areas involved in this quarter's efforts.

The assembly language version of the Kalman Filter tracking algorithm was completely recoded so that it would run in 32-bit floating point mode instead of 64-bit fixed point. These programs will handle the correlation of newly received data to locate the track table, start a new table for new targets, and move tables if required because of target changing drastically in azimuth and elevation. Also, they will perform the Bayesian estimation and will integrate the dynamic equations of motion over time for prediction and the coordinate conversions. These programs total, approximately, 4500 instructions and have been run on the timing simulator; however, they have not been run on the ILLIAC IV execution simulator because, at present, the simulator is not complete.

The main sections of the Kalman Filter programs-namely the Bayesian estimation and the integration of the dynamic equation of motion-have been programmed in ALGOL for the B5500 computer. These B5500 programs will be used to generate simulated tracking numbers for debugging the ILLIAC IV Kalman programs. The debugging of these programs will begin as soon as the execution simulator is operating. Also, the Tranquil version of the Kalman Filter tracking program will be debugged and run when the compiler and execution simulator are finished. By comparing the assembly language version and the Tranquil version of the program, it will give an evaluation of how efficiently these BMD problems can be programmed in the higher level language (Tranquil) for ILLIAC IV. This can test both the flexibility of debugging Tranquil programs over assembly language programs and the operating time penalty which is paid for using the higher level language. A report will be generated in the near future which will fully describe the Kalman Filter for ILLIAC IV.

Over this time period, there has been a slowdown in progress due to changes in personnel on this effort, and this slowdown will continue for the next time period. The two graduate students working on this effort have left the University. Presently, the effort has one full time professional, one part-time hourly, and one or two new graduate students who are just starting and require time to become familiar with ILLIAC IV.

A better understanding of the NISIM programs is being obtained in order to evaluate how this type of BMD programs would fit on ILLIAC IV. More analysis and a better understanding of them is still required before a complete sample BMD type problem can be set up for the ILLIAC IV computer.

5.12 ILLIAC IV Education

The ILLIAC IV Education for July and August consisted of organizing and presenting a course of instruction for ILLIAC IV personnel. Many newcomers were initially confused and swamped with the enormity of the ILLIAC IV project, but toward the end of the semester most of them were well oriented and producing useful work. The course was basically divided into three sections, and a brief discussion of each follows.

I. ALGOL and the B5500 -- The basic concepts of ALGOL and the special B5500 ALGOL features were presented. A detailed description of how to get programs running on the various B5500 hardware devices proved extremely valuable.

II. ILLIAC IV Assembler -- The Assembler was explained and examples given of the various types of orders, and case studies were examined. Several competent assembly language programmers emerged by the end of the summer.

III. Tranquil and ILLIAC IV Applications -- The Tranquil lectures had to be mostly theoretical because no facility for testing Tranquil code existed at the time of the course. General discussion of applications gave an insight into the scope and use of ILLIAC IV.

Work continued on compiling a useful programming manual for the Assembler. Parts of this have been completed and distributed to those requiring the information.

REFERENCES

[1] McCarthy, Thomas. Solution of Ordinary Differential Equations Related to Metabolic Systems. ILLIAC IV Document Number 197, (July 11, 1968).

[2] Winje, G. L. Implementation of the "Monte Carlo Evaluation of the Boltzmann Collision Integral" on ILLIAC IV. ILLIAC IV Document Number 200, (August 1, 1968).

[3] Nordsieck, Arnold, and Hicks, Bruce L. Monte Carlo Evaluation of the Boltzmann Collision Integral. Coordinated Science Laboratory Report R-307, (July, 1966).

[4] Faddeeva, V. N. <u>Computational Methods Of Linear Algebra</u>. New York: Dover Publications, Inc., 1959. p. 81 ff.

6. SWITCHING THEORY AND LOGICAL DESIGN

During the summer of 1968, C. R. Baugh, T. Ibaraki, T. K. Liu and S. Muroga had made significant progress in logical design of optimum networks by integer linear programming. Although our computational experience has been limited to switching functions of 3 variables, our logical design procedure by integer linear programming proved very efficient. When various restrictions such as maximum fan-ins and fan-outs are imposed on a network, logical design of an optimum network is very difficult and the only known design procedure is an exhaustive method (i.e. exhaust all network configurations and pick up the best one.) In 1963 Hellerman made the list of optimum network with NOR gates for all functions of 3 variables except the parity function $x_1 \oplus x_2 \oplus x_3$, because the parity function would take more than 1000 hours on an IBM 7090 by the exhaustive method. During the summer we found that our integer programming approach took 3 minutes on the IBM 360/75I to find an optimum network with NOR gates for the parity function. (4 minutes to exhaust all optimum networks for the function.)

Judging from our computational experience, our integer linear programming approach has the following features:

1. It is a very efficient method, though our computational experience is limited to functions of 3 variables and at most 8 gates.
2. Many parameters of a network such as the number of gates and the number of interconnections can be optimized.
3. Various network restrictions such as maximum fan-ins can be incorporated.
4. Incompletely specified functions can be treated by simply deleting the inequalities corresponding to the don't care vectors.

5. Optimum multiple output networks can also be designed.

6. Any types of gates can be treated. For example, NOR, NAND, AND, OR can be used, and optimum networks with a mixture of them can be designed. And also when a set of building blocks each of which is an aggregate of gates is given, optimum networks using these blocks can be designed.

7. The procedure can also be extended to the design of sequential networks. It has been known that when the number of states in a state transition diagram is minimized first (this is the standard design procedure of a sequential network in the conventional switching theory), the number of gates may not be minimized. The integer linear programming design method can design an optimum sequential network with the minimum number of gates, irrelevant of whether the number of states is minimized or not. Computation might be too time-consuming except for very simple cases.

8. The integer linear programming approach can be applied to diagnosis of a network (combinational and sequential).

9. An implicit enumeration algorithm is used in our integer linear programming design procedure with considerable modification by making use of intrinsic properties of network synthesis. The algorithm has at least the following two advantages. First, whenever we halt the computer due to an excess of computation time, the algorithm gives the best network available by then if any solution (which may prove to be optimum or not) has been found. Second, if any network is designed by some other means, the information can be utilized for the speed-up of our integer programming design procedure.

During the summer, E. Davidson completed excellent computer programs to design optimum networks with NOR gates, based upon the branch-bound method. His programs have a comparable speed with ours. However since his programs were made, by tailoring algorithm to NOR gates, without using inequalities, his programs appear more conplex than ours and are also difficult to change to other types of gates. Also it is difficult to apply his approach to the design of optimum sequential networks and diagnosis. Although his approach may have useful features, they are not obvious at this moment.

We are preparing reports on our results.

S. Muroga

7. REPORT OF THE STATISTICAL CONSULTANTS
July 1 through September 30, 1968

The activities of the Statistical Consultants may be classified under two headings: (1) consulting services to the statistical users of the Department facilities and (2) program development.

During the time period (July 1 - September 30) members of the group met with 245 different persons from 61 different departments to discuss computational problems connected with research projects. Since many persons returned more than once, there was a total of 1089 appointments over 64 working days, or 17 appointments per day.

Among the 245 research customers were 16 students with problems arising in course work, 24 students who were candidates for a Master's degree, and 56 students working on their doctoral theses.

The departments of the University with the most frequent requests are listed below:

DEPARTMENT	NUMBER OF PROJECTS	PER CENT OF TOTAL
Psychology	41	17.1
PE for Men	17	7.1
Political Science	17	7.1
Agricultural Economics	13	5.4
Education	13	5.4
Sociology	10	4.2
Educational Psychology	7	2.9
		49.2%

During the time period, 93 IBM 7094 SSUPAC Manuals were assembled and distributed.

Progress was made toward completion of SOUPAC (Statistically Oriented User's Programming and Consulting) for the IBM 360 system. The SOUPAC monitor was completed and checked. Except for minor revisions due to changes in the IBM 360 system programs, this section may be announced as completed.

The statistical user, however, has not yet been encouraged to begin to use SOUPAC. It seems sensible to wait until a few more subprograms are added to the system and until sets of instructions for the use of the subprograms have been duplicated. Otherwise, the user may find it necessary to alternate between the IBM 7094 and the IBM 360 to completed his project.

Kern Dickman

Kern W. Dickman
Project Director

(Supported in Part by the National Science Foundation under Grant No. NSF-GP-7634.)

8.1 New Routines - IBM System/360

E1-UØI-PØL1Z-78F-F

Polynomial Interpolation for a Specified Number of Points. Given a set of points $x_o,\ldots,x_{n-1}$ and the value of a function at these points $F(x_o),\ldots,F(x_{n-1})$, PØL1Z will calculate, for a given value y:

$$P(y) = F(x_i) + F'(x_i,x_{i+1})(y-x_i) + \ldots$$

$$+ F'(x_i,\ldots,x_j)(y-x_i)\ldots(y-x_{j-1})$$

where $F'(x_o,\ldots,x_k)$ denotes the k'th divided difference of F. The user specifies the number of points, $m = j-i+1$, to be used in the interpolation, and PØL1Z selects i so that $|y-x_i|$ and $|y-x_j|$ are minimized.

Programmed by Ruth Ann Fleck
July 10, 1968

E1-UØI-PØL2Z-79F-F

Polynomial Interpolation for a Specified Degree of Accuracy. Given a set of points $x_o,\ldots,x_{n-1}$ and the value of a function at these points $F(x_o),\ldots,F(x_{n-1})$, PØL2Z will calculate, for a given value y:

$$P(y) = F(x_i) + F'(x_i,x_{i+1})(y-x_i) + \ldots$$

$$+ F'(x_i,\ldots,x_j)(y-x_i)\ldots(y-x_{j-1})$$

where $F'(x_o,\ldots,x_k)$ denotes the k'th divided difference of F. The user specifies the accuracy he wishes to obtain and PØL2Z selects the number of points, $m = j-i+1$, to be used in the interpolation. i is chosen so that $|y-x_i|$ and $|y-x_j|$ are minimized.

Programmed by Ruth Ann Fleck
July 10, 1968

E1-UØI-DVDIFZ-80F-F

<u>Newton's Divided Differences for Use in Polynomial Interpolation</u>. Given a set of points $x_o,\ldots,x_{n-1}$ and the value of a function at these points $F(x_o)$, $\ldots,F(x_{n-1})$, DVDIFZ will calculate, for given values K and L, the K'th divided difference of F starting at location L, i.e.,

$$F'(x_L,x_{L+1},\ldots,x_{L+K}) =$$

$$\frac{(F'(x_{L+1},\ldots,x_{L+K}) - F'(x_L,\ldots,x_{L+K-1})}{x_{L+K} - x_L}$$

where $F'(x_L) = F(x_L)$. A parameter indicates which divided differences in the above recurrance relation have already been calculated. DVDIFZ was written primarily to be used internally by PØL1Z and PØL2Z.

Programmed by Ruth Ann Fleck
July 10, 1968

F2-UØI-SMEIGZ-81F-F

<u>Eigenvalues and Eigenvectors of a Symmetric Matrix</u>. Given a symmetric matrix S, SMEIGZ uses library subroutine HØUSEZ to reduce the matrix to tridiagonal form by Householder's method and then uses library subroutine CEST1Z to find the eigenvalues of this matrix, which are also the eigenvalues of S, by bisection. Library subroutine EVIITZ finds the eigenvectors of the tridiagonal matrix by inverse iteration and then library subroutine BKTRNZ transforms these eigenvectors into the eigenvectors of S.

Programmed by Ruth Ann Fleck
July 25, 1968

J5-UØI-LØGAXZ-84F-F

<u>CalComp Plotter Logarithmic Axis Routine</u>. This routine will draw a logarithmic axis containing a specified number of decades, label each decade with the appropriate power of 10, and put an EBCDIC label on the axis. This routine is adapted from the routine "LØGAX" from the Argonne National Laboratory.

Revised by H. G. Friedman, Jr.
July, 1968

J5-UØI-DASHZ-83F-F

<u>Dashed-Line Drawing Routine for the CalComp Plotter</u>. This routine will draw a dashed line between two given points. This routine is adapted from the routine "DASHLN" suggested in the IBM 7094 version of the CalComp SCOOP manual.

Revised by H. G. Friedman, Jr.
July, 1968

QO-UØI-TVECT-82M-A

<u>Transfer Vector Macro</u>. This is an assembly language macro which generates a series of branch instructions suitable for use as a transfer vector.

Written by H. G. Friedman, Jr.
July, 1968

8.2 Log Summaries

Table I - IBM 360/20

Summary of Use

July, 1968

Scheduled Engineering		34:47
Unscheduled Engineering		22:21
Maintenance		4:54
Production		186:30
Idle		94:25
Operator Training		9:08
	Total	352:05

Table II - IBM 360/20

Summary of Machine Errors

July, 1968

2560 MFCM		28
2203 Printer		2
	Total	30

Table I - IBM 360/50-75

Summary of Use

July, 1968

Unscheduled Engineering	42:50:00			
Scheduled Engineering	39:37:13			
Maintenance	20:39			
Idle	10:54:00			
			Subtotal	93:41:52
Total Use				
Training and Education		75:31:46		
System Improvement and Modifications		110:00:59		
System Updating		11:30:39		
University Administrative Overhead Use		10:36		
Customer Use				
In System	143:25:21			
Special Short Shots	12:27:27			
Customer Use Total		155:52:48		
			Total Use	353:06:48
			Total Time	446:48:40

Table II - IBM 360/50-75

Summary of Errors

July, 1968

2540 Card Reader Punch		3
75 CPU		5
50 CPU		3
Drum		1
1443 Printer		1
	Total Errors	13

IBM 360/75 TABLE III JULY, 1968

DEPT[1]	NUMBER OF RUNS			NUMBER OF SPECS			IBM 360/75 USAGE IN HH:MM:SS		
	T AND E[2]	RES	TOTAL	T AND E[2]	RES	TOTAL	T AND E[2]	RES	TOTAL
AAE	258	455	713	1	7	8	4:47:56	8:12:41	13: 0:37
ACCY	600	27	627	3	2	5	3:44:56	0:31:56	4:16:52
AGE	0	207	207	0	6	6	0: 0: 0	1:15: 7	1:15: 7
AGEC	11	191	202	1	7	8	0: 3:50	5:40:48	5:44:38
AGRON	0	231	231	0	5	5	0: 0: 0	1:52:38	1:52:38
ASTR	0	26	26	0	2	2	0: 0: 0	0:24:49	0:24:49
BECBSR	0	12	12	0	1	1	0: 0: 0	0: 7:32	0: 7:32
CE	1364	1469	2833	7	36	43	15:25:25	29:21: 5	44:46:30
CHE	130	753	883	1	25	26	1:51:21	10:42:56	12:34:17
CICBIO	0	34	34	0	1	1	0: 0: 0	0: 8:53	0: 8:53
CRC	0	11	11	0	1	1	0: 0: 0	0:14:27	0:14:27
CSL	0	24	24	0	2	2	0: 0: 0	0:43: 7	0:43: 7
DCS	5013	229	5242	4	10	14	39:41:55	13:56:51	53:38:46
DS	0	114	114	0	2	2	0: 0: 0	1:43:12	1:43:12
ECON	0	2	2	0	1	1	0: 0: 0	0:13:17	0:13:17
ED	0	37	37	0	2	2	0: 0: 0	1: 5:45	1: 5:45
EDADM	0	78	78	0	1	1	0: 0: 0	3: 1:11	3: 1:11
EE	171	396	567	2	20	22	2:14:55	4:35:38	6:50:33
ENGADM	0	56	56	0	2	2	0: 0: 0	0:29:35	0:29:35
ENGCST	0	4	4	0	1	1	0: 0: 0	0: 3:19	0: 3:19
FT	0	6	6	0	1	1	0: 0: 0	0: 3:41	0: 3:41
GEOG	0	6	6	0	1	1	0: 0: 0	0: 0:18	0: 0:18
GEOL	65	3	68	2	1	3	0:47:50	0: 0:17	0:48: 7
GSBA	18	0	18	1	0	1	2:37:31	0: 0: 0	2:37:31
HONORS	0	11	11	0	1	1	0: 0: 0	0:11:23	0:11:23
HORT	0	33	33	0	2	2	0: 0: 0	0:21:14	0:21:14
ICR	0	80	80	0	1	1	0: 0: 0	0:43:45	0:43:45
ILLDMH	0	321	321	0	1	1	0: 0: 0	6:58: 2	6:58: 2
INADM	16	8	24	1	1	2	0:17:20	0: 3: 6	0:20:26
MATH	0	8	8	0	2	2	0: 0: 0	0: 0:57	0: 0:57
MATRL	0	329	329	0	13	13	0: 0: 0	8:35:11	8:35:11
MCBIO	0	1	1	0	1	1	0: 0: 0	0: 0:27	0: 0:27
ME	124	14	138	3	2	5	0:33:44	0:11: 5	0:44:49
MMPE	0	12	12	0	2	2	0: 0: 0	0:23:26	0:23:26
MUSIC	6	0	6	1	0	1	0: 2:34	0: 0: 0	0: 2:34
NHS	0	19	19	0	2	2	0: 0: 0	0:10:59	0:10:59
NUCE	17	208	225	1	6	7	0: 7:19	3:39:44	3:47: 3

OIR	0	109	109	0	1	1	0: 0: 0	2:32:25	2:32:25
PHYB	0	10	10	0	1	1	0: 0: 0	0:14:57	0:14:57
PHYCS	0	535	535	0	10	10	0: 0: 0	7:32:43	7:32:43
PHYX	0	568	568	0	27	27	0: 0: 0	8:31:14	8:31:14
POLS	45	4	49	1	1	2	2:49:43	0:10:30	2:39:13
PSYCH	2	84	86	1	3	4	0: 0:54	0:51:19	0:52:13
REFUND	0	21	21	0	1	1	0: 0: 0	1:35:52	1:35:52
SCONS	0	31	31	0	1	1	0: 0: 0	0:36:17	0:36:17
SGS	0	76	76	0	2	2	0: 0: 0	0:41:56	0:41:56
SOC	0	1	1	0	1	1	0: 0: 0	0: 2:34	0: 2:34
SRL	0	79	79	0	3	3	0: 0: 0	1:12:16	1:12:16
SWS	0	128	128	0	4	4	0: 0: 0	3:43: 4	3:43: 4
TAM	31	341	372	2	11	13	0:24:33	7:25: 8	7:49:41
VPH	0	1	1	0	1	1	0: 0: 0	0: 0:27	0: 0:27
VPP	0	6	6	0	1	1	0: 0: 0	0: 1:52	0: 1:52
WPGU	0	4	4	0	1	1	0: 0: 0	0: 1:55	0: 1:55
ZOOL	0	326	326	0	9	9	0: 0: 0	2:43:30	2:43:30
SUBTOTAL	7871	7739	15610	32	249	281	75:31:46	143:25:21	218:57: 7
DCSSYS[3]	0	2709	2709	0	22	22	0: 0: 0	110: 0:59	110: 0:59
SSUAD[4]	0	25	25	0	1	1	0: 0: 0	0:10:36	0:10:36
XDCS[5]	0	46	46	0	3	3	0: 0: 0	11:30:39	11:30:39
XMAINT[6]	C	10	10	0	4	4	0: 0: 0	93:41:52	93:41:52
XSSS[7]	0	653	653	0	1	1	0: 0: 0	12:27:27	12:27:27
TOTAL	7871	11182	19053	32	280	312	75:31:46	371:16:54	446:48:40

1 See list of departmental codes following
2 Training and Education
3 System Improvement and Modifications
4 University Administrative Overhead Use
5 System Updating
6 System Maintenance
7 Special Shorts Shots

Table I - IBM 360/20

Summary of Use

August, 1968

Scheduled Engineering	11:40
Unscheduled Engineering	17:00
Maintenance	4:32
Production	276:30
Code Check	2:05
Operator Training	2:11
Idle	39:54
Total	353:52

Table II - IBM 360/20

Summary of Machine Errors

August, 1968

2560 MFCM	18
2203 Printer	7
2020 Processing Unit	2
Total	27

Table I - IBM 360/50-75

Summary of Use

August, 1968

Unscheduled Engineering	9:36:00			
Scheduled Engineering	63:57:46			
Maintenance	15:39			
Idle	15:09:00			
			Subtotal	88:58:25
Total Use				
Training and Education		57:11:55		
System Improvement and Modifications		83:13:30		
System Updating		13:48:11		
University Administrative Overhead Use		1:20:00		
Customer Use				
In System	162:33:31			
Special Short Shots	13:33:48			
Customer Use Total		176:07:19		
			Total Use	331:40:55
			Total Time	420:39:20

Table II - IBM 360/50-75

Summary of Errors

August, 1968

2540 Card Reader Punch		1
50 CPU		2
	Total Errors	3

IBM 360/75 TABLE III AUGUST,1963

DEPT[1]	NUMBER OF RUNS			NUMBER OF SPECS			IBM 360/75 USAGE IN HH:MM:SS		
	T AND E[2]	RES	TOTAL	T AND E[2]	RES	TOTAL	T AND E[2]	RES	TOTAL
AAE	152	549	701	1	10	11	4:13: 7	10:25:39	14:38:46
ACCY	515	14	529	2	1	3	4: 4:48	0:20:18	4:25: 6
AGE	0	246	246	0	6	6	0: 0: 0	2: 8:23	2: 8:23
AGEC	0	410	410	0	7	7	0: 0: 0	12:24:49	12:24:49
AGRON	0	192	192	0	5	5	0: 0: 0	1:41:46	1:41:46
ANS	0	5	5	0	1	1	0: 0: 0	0: 0:41	0: 0:41
ASTR	0	7	7	0	2	2	0: 0: 0	0: 8:28	0: 8:28
BECBSR	0	3	3	0	1	1	0: 0: 0	0: 1: 5	0: 1: 5
CE	792	1403	2195	5	42	47	12:45:11	31:17:58	44: 3: 9
CHE	10	541	551	1	24	25	0: 5:34	14: 5:30	14:11: 4
CICBIO	0	40	40	0	1	1	0: 0: 0	0:56:20	0:56:20
CRC	0	4	4	0	1	1	0: 0: 0	0: 2:18	0: 2:18
CSL	0	39	39	0	2	2	0: 0: 0	2:23:18	2:23:18
DCS	2517	266	2783	4	8	12	28:21:43	10:11:44	38:33:27
DS	0	50	50	0	2	2	0: 0: 0	0:53:17	0:53:17
ECON	0	3	3	0	1	1	0: 0: 0	0:15:22	0:15:22
ED	0	26	26	0	2	2	0: 0: 0	0:15: 4	0:15: 4
EDADM	0	59	59	0	1	1	0: 0: 0	2:44:41	2:44:41
EE	10	569	579	1	22	23	0: 4:15	7: 2:34	7: 6:49
ENGADM	0	36	36	0	1	1	0: 0: 0	0:13:39	0:13:39
ENGCST	0	32	32	0	1	1	0: 0: 0	0:43: 9	0:43: 9
FT	0	24	24	0	1	1	0: 0: 0	0:11:24	0:11:24
GEOG	0	6	6	0	1	1	0: 0: 0	0: 0:57	0: 0:57
GEOL	63	88	151	2	1	3	0:38:15	1:17:39	1:55:54
GSBA	36	0	36	1	0	1	1:41:31	0: 0: 0	1:41:31
HEC	0	23	23	0	1	1	0: 0: 0	0: 8:11	0: 8:11
HONORS	0	1	1	0	1	1	0: 0: 0	0: 1:19	0: 1:19
HORT	0	44	44	0	3	3	0: 0: 0	0:20:46	0:20:46
ICR	0	40	40	0	1	1	0: 0: 0	0:30:31	0:30:31
ILLDMH	0	303	303	0	1	1	0: 0: 0	15:36:44	15:36:44
INADM	51	1	52	1	1	2	0:54: 1	0: 0:12	0:54:13
MATH	0	1	1	0	1	1	0: 0: 0	0: 0:16	0: 0:16
MATRL	0	364	364	0	19	19	0: 0: 0	4:34:38	4:34:38
MCBIO	0	25	25	0	1	1	0: 0: 0	0: 9:41	0: 9:41
ME	14	12	26	3	1	4	0: 3:37	0: 5:29	0: 9: 6
MMPE	0	26	26	0	1	1	0: 0: 0	0:15:40	0:15:40
MUSIC	39	0	39	1	0	1	0:29:55	0: 0: 0	0:29:55

NHS	0	5	5	0	1	1	0: 0: 0	0: 1:26	0: 1:26
NUCE	85	129	214	2	6	8	0:52:40	1:57:29	2:50: 9
OIR	0	126	126	0	1	1	0: 0: 0	2: 3:30	2: 3:30
PHYB	0	92	92	0	2	2	0: 0: 0	1:17: 1	1:17: 1
PHYCS	0	459	459	0	10	10	0: 0: 0	7:29:17	7:29:17
PHYX	0	511	511	0	6	6	0: 0: 0	17:44: 8	17:44: 8
PLPA	1	0	1	1	0	1	0: 0: 8	0: 0: 0	0: 0: 8
POLS	50	0	50	1	0	1	2: 6: 0	0: 0: 0	2: 6: 0
PSYCH	40	58	98	1	4	5	0:27:49	0:36: 8	1: 3:57
REC	0	20	20	0	1	1	0: 0: 0	0:40:40	0:40:40
REFUND	0	28	28	0	1	1	0: 0: 0	0:36:28	0:36:28
SCONS	0	19	19	0	1	1	0: 0: 0	0:17: 0	0:17: 0
SGS	0	105	105	0	2	2	0: 0: 0	0:55:25	0:55:25
SRL	0	46	46	0	3	3	0: 0: 0	0:51: 3	0:51: 3
SWS	0	87	87	0	6	6	0: 0: 0	1:19:13	1:19:13
TAM	24	459	483	2	12	14	0:23:21	4:37: 7	5: 0:28
VPP	0	2	2	0	1	1	0: 0: 0	0: 0:46	0: 0:46
WPGU	0	4	4	0	1	1	0: 0: 0	0: 1:57	0: 1:57
ZOOL	0	129	129	0	7	7	0: 0: 0	1: 6: 7	1: 6: 7
SUBTOTAL	4399	7786	12185	29	240	269	57:11:55	162:33:31	219:45:26
DCSSYS[3]	0	2859	2859	0	20	20	0: 0: 0	33:13:30	33:13:30
SSUAD[4]	0	27	27	0	1	1	0: 0: 0	1:20: 0	1:20: 0
XDCS[5]	0	60	60	0	3	3	0: 0: 0	13:48:11	13:48:11
XMAINT[6]	0	23	23	0	4	4	0: 0: 0	88:58:25	88:58:25
XSSS[7]	0	667	667	0	1	1	0: 0: 0	13:33:48	13:33:48
TOTAL	4399	11422	15821	29	269	298	57:11:55	363:27:25	420:39:20

1 See list of departmental codes following
2 Training and Education
3 System Improvement and Modifications
4 University Administrative Overhead Use
5 System Updating
6 System Maintenance
7 Special Short Shots

Table I - IBM 360/20

Summary of Use

September, 1968

Unscheduled Engineering		35:32
Scheduled Engineering		7:09
Maintenance		3:33
Production		227:34
Code Check		4:30
Operator Training		:07
Air Conditioning		16:47
Idle		31:05
	Total	326:17

Table II - IBM 360/20

Summary of Machine Errors

September, 1968

2560 MFCM		7
2020 Processing Unit		1
	Total	8

Table I - IBM 360/50-75

Summary of Use

September, 1968

Unscheduled Engineering	17:15:00		
Scheduled Engineering	43:24:43		
Power Failure	1:10:00		
Air Conditioning	32:30:00		
Idle	9:10:00		
Maintenance	:12		
		Subtotal	103:29:55
Total Use			
Training and Education		24:25:53	
System Improvement and Modifications		192:54:31	
System Updating		20:28:40	
University Administrative Overhead Use		56:00	
Overhead		28:38	
Refund		:20	
Customer Use			
In System	95:30:33		
Special Short Shots	3:21:18		
Customer Use Total		98:51:51	
		Total Use	338:05:53
		Total Time	441:35:48

Table II - IBM 360/50-75

Summary of Errors

September, 1968

75 Core		2
	Total	2

IBM 360/75 TABLE III SEPT., 1968

DEPT[1]	NUMBER OF RUNS T AND E[2]	NUMBER OF RUNS RES	NUMBER OF RUNS TOTAL	NUMBER OF SPECS T AND E[2]	NUMBER OF SPECS RES	NUMBER OF SPECS TOTAL	IBM 360/75 USAGE IN HH:MM:SS T AND E[2]	IBM 360/75 USAGE IN HH:MM:SS RES	IBM 360/75 USAGE IN HH:MM:SS TOTAL
AAE	495	144	639	4	9	13	7: 6:22	1:44:49	8:51:11
ACCY	88	98	186	3	2	5	0:21:32	0:44: 9	1: 5:41
AGE	52	110	162	3	5	8	0:51:23	0:53:23	1:44:46
AGEC	0	122	122	0	6	6	0: 0: 0	3:34:16	3:34:16
AGRON	41	132	173	1	5	6	0: 9:52	0:51:54	1: 1:46
ANS	0	46	46	0	1	1	0: 0: 0	0: 9:47	0: 9:47
ARCH	36	0	36	2	0	2	0: 5:21	0: 0: 0	0: 5:21
ASTR	0	6	6	0	3	3	0: 0: 0	0:10: 0	0:10: 0
EECBSR	0	4	4	0	1	1	0: 0: 0	0: 1:17	0: 1:17
CE	158	595	753	6	36	42	0: 6:45	12:29:48	12:36:33
CHE	7	454	461	2	20	22	0: 1:52	10: 8:25	10:10:17
CICBIO	0	0	0	0	1	1	0: 0: 0	0:55:14	0:55:14
CRC	0	18	18	0	1	1	0: 0: 0	0:17:14	0:17:14
CSL	0	3	3	0	1	1	0: 0: 0	0: 0:47	0: 0:47
DCS	2684	128	2812	13	8	21	9:11:46	3:19:49	12:31:35
DS	0	49	49	0	2	2	0: 0: 0	3: 8:47	3: 8:47
ECON	0	10	10	0	1	1	0: 0: 0	0:17:45	0:17:45
ED	0	10	10	0	2	2	0: 0: 0	0:18:10	0:18:10
EDADM	0	8	8	0	1	1	0: 0: 0	0:16:56	0:16:56
EE	2	486	488	1	24	25	0: 0:12	7:26: 9	7:26:21
ENGADM	0	62	62	0	2	2	0: 0: 0	0:37:31	0:37:31
ENGCST	0	56	56	0	1	1	0: 0: 0	0:40:57	0:40:57
ENGH	51	0	51	1	0	1	0: 5:13	0: 0: 0	0: 5:13
FT	0	39	39	0	2	2	0: 0: 0	0: 8:29	0: 8:29
GENE	1	0	1	1	0	1	0: 0:56	0: 0: 0	0: 0:56
GEOG	0	3	3	0	1	1	0: 0: 0	0: 0:57	0: 0:57
GEOL	8	1	9	1	1	2	0: 4:31	0: 0: 8	0: 4:39
GER	0	7	7	0	1	1	0: 0: 0	0:10:32	0:10:32
GSBA	165	0	165	2	0	2	5: 9: 1	0: 0: 0	5: 9: 1
HORT	0	15	15	0	1	1	0: 0: 0	0:19:51	0:19:51
ICR	0	51	51	0	1	1	0: 0: 0	0:48: 9	0:48: 9
ILLCF	0	7	7	0	1	1	0: 0: 0	0: 2:17	0: 2:17
ILLDMH	0	290	290	0	1	1	0: 0: 0	10:36:40	10:36:40
INADM	0	43	43	0	1	1	0: 0: 0	0:58:32	0:58:32
MATH	0	24	24	0	2	2	0: 0: 0	0:20:35	0:20:35
MATRL	0	306	306	0	18	18	0: 0: 0	5:16:17	5:16:17
MCBIO	0	20	20	0	1	1	0: 0: 0	0: 5:55	0: 5:55
ME	278	34	312	6	3	9	1:22:10	0:29:55	1:52: 5
MMPE	0	59	59	0	2	2	0: 0: 0	0:41:13	0:41:13
MODINE	0	5	5	0	1	1	0: 0: 0	0: 0:42	0: 0:42
MUSIC	0	1	1	0	1	1	0: 0: 0	0: 0: 7	0: 0: 7
NHS	0	29	29	0	1	1	0: 0: 0	0:31:29	0:31:29
NUCE	14	30	44	2	3	5	0: 3:13	0:51:40	0:54:53

CIR	0	82	82	0	1	1	0: 0: 0	1:20:56	1:20:56
CVRHED	0	203	203	0	1	1	0: 0: 0	0:28:38	0:28:38
PHYB	0	94	94	0	2	2	0: 0: 0	1:33:35	1:33:35
PHYCS	0	512	512	0	13	13	0: 0: 0	9: 5:18	9: 5:18
PHYX	0	551	551	0	5	5	0: 0: 0	11:34:11	11:34:11
PCLS	0	0	0	1	0	1	0:30:10	0: 0: 0	0:30:10
PSYCH	18	3	21	1	2	3	0:15:35	0: 1: 6	0:16:41
REC	0	4	4	0	1	1	0: 0: 0	0:15:24	0:15:24
SCCNS	0	41	41	0	1	1	0: 0: 0	0:29:17	0:29:17
SGS	0	74	74	0	2	2	0: 0: 0	0:44:48	0:44:48
SRL	0	60	60	0	3	3	0: 0: 0	0:38:56	0:38:56
SWS	0	38	38	0	4	4	0: 0: 0	0:32: 7	0:32: 7
TAM	0	123	123	0	8	8	0: 0: 0	1:44:27	1:44:27
LSGS	0	1	1	0	1	1	0: 0: 0	0: 0:52	0: 0:52
VPH	0	14	14	0	1	1	0: 0: 0	0: 6:34	0: 6:34
VTED	0	1	1	0	1	1	0: 0: 0	0: 0: 3	0: 0: 3
WPGU	0	9	9	0	2	2	0: 0: 0	0: 3:59	0: 3:59
ZCCL	2	28	30	1	6	7	0: 0:19	0:14:23	0:14:42
SUBTOTAL [3]	4100	5343	9443	51	228	279	24:25:53	95:59:11	120:25: 4
DCSSYS [4]	0	2859	2899	0	22	22	0: 0: 0	192:54:31	192:54:31
REFUND [5]	0	8	8	0	1	1	0: 0: 0	0: 0:20	0: 0:20
SSUAD [6]	0	34	34	0	1	1	0: 0: 0	0:56: 0	0:56: 0
XCCS [7]	0	114	114	0	2	2	0: 0: 0	20:28:40	20:28:40
XMAINT	0	16	16	0	6	6	0: 0: 0	103:29:55	103:29:55
XSSS [8]	0	288	288	0	1	1	0: 0: 0	3:21:18	3:21:18
TOTAL	4100	8702	12802	51	261	312	24:25:53	417: 9:55	441:35:48

1 See list of departmental codes following
2 Training and Education
3 System Improvement and Modifications
4 Rerun of User Jobs Due to Machine or System Error
5 University Administrative Overhead Use
6 System Updating
7 System Maintenance
8 Special Short Shots

IBM 360/75 TABLE III 3RD QTR, 1968

DEPT[1]	NUMBER OF RUNS			NUMBER OF SPECS			IBM 360/75 USAGE IN HH:MM:SS		
	T AND E[2]	RES	TOTAL	T AND E[2]	RES	TOTAL	T AND E[2]	RES	TOTAL
AAE	905	1148	2053	6	26	32	16: 7:25	20:23: 9	36:30:34
ACCY	1203	139	1342	8	5	13	8:11:16	1:36:23	9:47:39
AGE	52	563	615	3	17	20	0:51:23	4:16:53	5: 8:16
AGEC	11	723	734	1	20	21	0: 3:50	21:39:53	21:43:43
AGRON	41	555	596	1	15	16	0: 9:52	4:26:18	4:36:10
ANS	0	51	51	0	2	2	0: 0: 0	0:10:28	0:10:28
ARCH	36	0	36	2	0	2	0: 5:21	0: 0: 0	0: 5:21
ASTR	0	39	39	0	7	7	0: 0: 0	0:43:17	0:43:17
BECBSR	0	19	19	0	3	3	0: 0: 0	0: 9:54	0: 9:54
CE	2314	3467	5781	18	114	132	28:17:21	73: 8:51	101:26:12
CHE	147	1748	1895	4	69	73	1:58:47	34:56:51	36:55:38
CICBIO	0	74	74	0	3	3	0: 0: 0	0: 9:59	0: 9:59
CRC	0	33	33	0	3	3	0: 0: 0	0:33:59	0:33:59
CSL	0	116	116	0	5	5	0: 0: 0	3: 7:12	3: 7:12
DCS	10214	623	10837	21	26	47	77:15:24	27:28:24	104:43:48
DS	0	213	213	0	6	6	0: 0: 0	5:45:16	5:45:16
ECON	0	15	15	0	3	3	0: 0: 0	0:19:50	0:19:50
ED	0	73	73	0	6	6	0: 0: 0	1:38:59	1:38:59
EDADM	0	145	145	0	3	3	0: 0: 0	6: 2:48	6: 2:48
EE	183	1451	1634	4	66	70	2:19:22	19: 4:21	21:23:43
ENGADM	0	154	154	0	5	5	0: 0: 0	1:20:45	1:20:45
ENGCST	0	92	92	0	3	3	0: 0: 0	1:27:25	1:27:25
ENGH	51	0	51	1	0	1	0: 5:13	0: 0: 0	0: 5:13
FT	0	69	69	0	4	4	0: 0: 0	0:23:34	0:23:34
GENE	1	0	1	1	0	1	0: 0:56	0: 0: 0	0: 0:56
GEOG	0	15	15	0	3	3	0: 0: 0	0: 2:12	0: 2:12
GEOL	136	92	228	5	3	8	1:30:36	1:18: 4	2:48:40
GER	0	7	7	0	1	1	0: 0: 0	0:10:32	0:10:32
GSBA	219	0	219	4	0	4	9:28: 3	0: 0: 0	9:28: 3
HEC	0	28	28	0	1	1	0: 0: 0	0: 8:11	0: 8:11
HONORS	0	12	12	0	2	2	0: 0: 0	0:12:42	0:12:42
HORT	0	92	92	0	6	6	0: 0: 0	1: 1:51	1: 1:51
ICR	0	171	171	0	3	3	0: 0: 0	2: 2:25	2: 2:25
ILLCF	0	7	7	0	1	1	0: 0: 0	0: 2:17	0: 2:17
ILLDMH	0	914	914	0	3	3	0: 0: 0	33:11:26	33:11:26
INADM	67	52	119	2	3	5	1:11:21	1: 1:50	2:13:11

MATH	0	33	33	0	5	5	0: 0: 0	0:21:48	0:21:48
MATRL	0	999	999	0	50	50	0: 0: 0	18:26: 6	18:26: 6
MCBIO	0	46	46	0	3	3	0: 0: 0	0:16: 3	0:16: 3
ME	416	60	476	12	6	18	1:59:31	0:46:29	2:46: 0
MMPE	0	97	97	0	5	5	0: 0: 0	1:20:19	1:20:19
MODINE	0	5	5	0	1	1	0: 0: 0	0: 0:42	0: 0:42
MUSIC	45	1	46	2	1	3	0:32:29	0: 0: 7	0:32:36
NHS	0	53	53	0	4	4	0: 0: 0	0:43:54	0:43:54
NUCE	116	367	483	5	15	20	1: 3:12	6:28:53	7:32: 5
OIR	0	317	317	0	3	3	0: 0: 0	5:56:51	5:56:51
PHYB	0	196	196	0	5	5	0: 0: 0	3: 5:33	3: 5:33
PHYCS	0	1506	1506	0	33	33	0: 0: 0	24: 7:18	24: 7:18
PHYX	0	1630	1630	0	38	38	0: 0: 0	37:49:33	37:49:33
PLPA	1	0	1	1	0	1	0: 0: 8	0: 0: 0	0: 0: 8
POLS	95	4	99	3	1	4	4:25:33	0:10:30	4:15: 3
PSYCH	60	145	205	3	9	12	0:44:18	1:28:33	2:12:51
REC	0	24	24	0	2	2	0: 0: 0	0:56: 4	0:56: 4
SCONS	0	91	91	0	3	3	0: 0: 0	1:22:34	1:22:34
SGS	0	255	255	0	6	6	0: 0: 0	2:22: 9	2:22: 9
SOC	0	1	1	0	1	1	0: 0: 0	0: 2:34	0: 2:34
SRL	0	185	185	0	9	9	0: 0: 0	2:42:15	2:42:15
SWS	0	253	253	0	14	14	0: 0: 0	5:34:24	5:34:24
TAM	55	923	978	4	31	35	0:47:54	13:46:42	14:34:36
USGS	0	1	1	0	1	1	0: 0: 0	0: 0:52	0: 0:52
VPH	0	15	15	0	2	2	0: 0: 0	0: 7: 1	0: 7: 1
VPP	0	8	8	0	2	2	0: 0: 0	0: 2:38	0: 2:38
VTED	0	1	1	0	1	1	0: 0: 0	0: 0: 3	0: 0: 3
WPGU	0	17	17	0	4	4	0: 0: 0	0: 7:51	0: 7:51
ZOOL	2	483	485	1	22	23	0: 0:19	4: 4: 0	4: 4:19
SUBTOTAL	16370	20616	36986	112	714	826	157: 9:34	399:17:05	556:26:39

DCSSYS[3]	0	8467	8467	0	64	64	0: 0: 0	386: 9: 0	386: 9: 0
OVRHED[4]	0	203	203	0	1	1	0: 0: 0	0:28:38	0:28:38
REFUND[5]	0	57	57	0	3	3	0: 0: 0	2:12:40	2:12:40
SSUAD[6]	0	86	86	0	3	3	0: 0: 0	2:26:36	2:26:36
XUCS[7]	0	22[illegible]	22[illegible]	0	8	8	0: 0: 0	45:47:30	45:47:30
XMAINT[8]	0	49	49	0	14	14	0: 0: 0	286:10:12	286:10:12
XSSS[9]	0	1608	1608	0	3	3	0: 0: 0	29:22:33	29:22:33
TOTAL	16370	31306	47676	112	810	922	157: 9:34	1151:54:14	1309: 3:48

1 See list of departmental codes following
2 Training and Education
3 System Improvement and Modifications
4 System Use in Excess of User Estimated Usage
5 Rerun of User Jobs Due to Machine or System Error
6 University Administrative Overhead Use
7 System Updating
8 System Maintenance
9 Special Short Shots

8.3 Research Problem Specifications

During the third quarter of 1968, 138 problem specifications were submitted to the Department for computation on the System/360. The following brief descriptions of these problems have been prepared for inclusion in this report by those submitting them. T indicates a calculation associated with a thesis.

1660 Agricultural Economics. Economic Analysis of Water Use in Illinois Agriculture. Estimates will be made of the change that occurs in average returns per acre of corn when irrigation is adopted. Estimates of the change in year-to-year variations will also be made. The sources of corn-yield data will be both commercial farm yields and experimental yields. Temperature and rainfall data will be used to explain corn yield variation and to form the basis for expected yields under irrigation. The principal statistical techniques used will be multiple regression, with corn yield as the dependent variable and management practices, temperature, and rainfall as independent variables. (E. R. Swanson)

1661 T Civil Engineering. Analysis of Plates and Shell by Integral Equation Methods. The solutions to the boundary value problems of potential theory, elastic plate and shell theories are formulated in terms of internal equations by using certain singular fundamental solutions of the governing differential equations in the reciprocal formula for the structure. The integral equations are approximated by a set of algebraic equations which are solved on the computer. (Delroy Forbes)

1663 T Electrical Engineering. State-Dependent Time Delay Systems. It is desired to determine the optimal control of systems with state-dependent time delays by numerical methods for a class of linear systems. (Tomko)

LIST OF DEPARTMENT CODES

If your department or office does not appear on this list, please write its full name in the department field on the Problem Specification Form even though it will require more than 6 characters.

ACCY	Accountancy
ADMREC	Admissions and Records
ADV	Advertising
AAE	Aeronautical and Astronautical Eng.
AGEC	Agricultural Economics
AGE	Agricultural Engineering
AGREXT	Agricultural Extension
AGRON	Agronomy
ANS	Animal Science
ANTH	Anthropology
ARCH	Architecture
ASTR	Astronomy
BIOPH	Biophysics
BOT	Botany
BCMPL	Bureau of Community Planning
BECBSR	Bureau of Economic Business Research
BEDRES	Bureau of Educational Research
BINRES	Bureau of Institutional Research
BED	Business Education
CZR	Center for Zoonoses Research
CERE	Ceramic Engineering
CPUBS	Champaign Public Schools
CHE	Chemistry and Chemical Engineering
CRC	Children's Research Center
CP	City Planning
CE	Civil Engineering
COMM	Communications
CURLAB	Curriculum Laboratory
DS	Dairy Science
DCS	Department of Computer Science
DGS	Division of General Studies
DUE	Division of University Extension
DOW	Division of Waterways
ECON	Economics
ED	Education
EDPSY	Educational Psychology
EDADM	Educational Admin. and Supervision
EDTEST	Educational Testing
EE	Electrical Engineering
ENGADM	Engineering Administration
ENGCST	Engineering College and Station
ENGH	Engineering Honors Program
ENGLSH	English
ENTOM	Entomology
EDC	Extension Division Counseling
FIN	Finance
FT	Food Science
FOR	Forestry
GENE	General Engineering
GEOG	Geography
GEOL	Geology
GER	German
GSBA	Graduate School of Business Admin.
HED	Health Education
HLTHSV	Health Service
HEC	Home Economics
HONORS	Honors Program
HORT	Horticulture
ILLDMH	Illinois Dept. of Mental Health
INADM	Industrial Administration
IE	Industrial Engineering
IREC	Institute for Researh on Exc. Children
ICR	Institute of Communications Research
IGPA	Institute of Govt. and Public Affairs
ILR	Institute of Labor and Ind. Relations
LIBS	Library Science
LING	Linguistics
MKTG	Marketing
MATRL	Materials Research Laboratory
MATH	Mathematics
ME	Mechanical Engineering
MCBIO	Microbiology
MMPE	Mining, Metallurgy, and Petroleum Eng.
MUSIC	Music
NHS	Natural History Survey
NUCE	Nuclear Engineering
OAC	Office of Agricultural Communication
DNW	Office of the Dean of Women
OIR	Office of Instructional Resources
PEM	PE for Men and Graduate PE
PEW	PE for Women
PHYPLA	Physical Plant
PHYCS	Physics
PHYB	Physics Betatron Laboratory
PHYX	Physics Project X
PHYSL	Physiology and Biophysics
PLPA	Plant Pathology
POLS	Political Science
PROVST	Provost's Office
PSYCH	Psychology
REC	Recreation
SHCBRC	Small Homes Council
SOCW	Social Work
SOC	Sociology
SCONS	Soil Conservation Service
SPED	Special Education
PSDEC	Special Education, Decatur Pub. Schools
SPCH	Speech and Theatre
SGS	State Geological Survey
SWS	State Water Survey
SCS	Student Counseling Service
SRL	Survey Research Laboratory
TAM	Theoretical and Applied Mechanics
USGS	U.S. Geological Service
UNIHI	University High School
UCCTE	Urbana-Champaign Coun. on Teacher Ed.
VMS	Veterinary Medical Science
VMA	Veterinary Medicine Administration
VPH	Veterinary Pathology and Hygiene
VPP	Veterinary Physiology and Pharmacology
VTED	Vocational and Technical Education
WPGU	WPGU Radio Station
ZOOL	Zoology

Chicago Circle

CCCHE	Chemistry
CCDME	Materials Engineering
CCENEN	Energy Engineering
CCPHCS	Physics
CCSOC	Sociology
CCSCS	Student Counseling Service

Medical Center

ORME	Office of Research in Medical Ed.
OT	Occupational Therapy

Ill. State University

ISEDAD	Department of Education Administration

1667 T Chemistry and Chemical Engineering. Semi-Empirical M. O. Calculation. Application of semi-empirical molecular orbital theory to the description of carbonyl complexes. (T. L. Brown)

1668 Veterinary Pathology and Hygiene. Micrometeorology. Data on micrometeorological conditions and survival and development of infective nematode larvae are collected at the Veterinary Research Farm. Computers are used for data reduction and statistical analyses. (K. S. Todd)

1669 Soil Conservation Service. Hydraulic, Hydrology and Economic Analysis. Using Manning's formula, the hydraulic properties of a stream outlet for a watershed will be determined. (USDA SCS)

1670 Accountancy. Elementary Programs to Understand PL/I Language. The problems to be submitted will be of an elementary nature in order to obtain a basic knowledge of PL/I. (Michael Moebs)

1673 Mechanical Engineering. Minor Heat Transfer Problems. This is the same as problem specification 37038 on the 7094. (L. D. Savage)

1674 Mechanical Engineering. Computer Workshop in Air Conditioning Design. Design problems in air conditioning. (Stoecker)

1675 Psychology. Adult and Child Objective Test Battery. The history and effects of personality factors, and the objective measurement of their dynamic structure. (Cattell)

1678 Natural History Survey. Effects of Chelating Agents and Metal Chelates on Insects. The toxicological and physiological effects of chelating agents and chelated metals, especially copper (II) are being studied. Such parameters as mortality, fecundity, fertility, and metabolic abnormalities are utilized. (Douglas K. Sell)

1679 U. S. Geological Survey. Hydrologic Investigations. This replaces problem specification 5D039. (USGS WRD)

1680 Computer Science. ILLIAC IV. For use of 360/20 for reproducing, interpreting, etc. (Nancy K. Stone)

1681 Materials Research Laboratory. Thermal Conductivity of Helium. This computation will be done to give the thermal conductivity of solid helium containing isotopic impurities. As an intermediate calculation, phonon scattering rates will be computed with the computer. (Hilary D. Jones)

1682 Electrical Engineering. Stepping Motor Simulation. Digital similation of stepping motor to obtain an optimal response under various conditions. (B. C. Kuo)

1683 School of Life Sciences. Speciation in Western Mammals. The species-status of two populations of pocket gophers is being tested. Problems in speciation and numerical taxonomy are involved. (Hoffmeister)

1684 T Computer Science. Modeling of Domain Growth Activity in Polycrystalline Ferrites. The objective of this problem is to study the effects of the magnetic parameters of polycrystalline ferrites on the time rate of change of flux in these ferrites. This will be accomplished by simulating the movement of the magnetic domain boundaries which occurs in a polycrystalline ferrite. The flux curves obtained will be compared to experimentally obtained curves. (R. P. Harms)

1685 T Chemistry and Chemical Engineering. The Effect of Vibrational-Translational Energy Transfer on Reaction Rate. Using action-angle formalism to calculate the effect of vibrational-translational energy transfer on the reaction cross section. (Shiou-Fu Wū)

1686 T Zoology. Niche Segregation Among Birds of Trelease Woods. This is a study of niche segregation and habitat utilization among several species of birds at Trelease Woods. (Robert E. Lewke)

1687 T Coordinated Science Laboratory. Cognitive Heuristic Problem Solver. The thesis explores the problem of writing heuristic programs whose rules of thumb are adopted and improved by the program itself, based on the recognition of relevant problem features, possibly aided by interaction with a human operator. The problem to be solved by the program is the combinational layout of connected 2-dimensional objects, with minimum connection cost as objective. (Bielowski)

1689 Materials Research Laboratory. Electro-Absorption. An investigation of the band structure of semiconductors from their optical properties. (Paul Handler)

1690 Chemistry and Chemical Engineering. Fourier Transform of Pulsed NMR Spectra. This computation time will be used to calculate, and plot the fourier transform of a function determined by the data output of a Pulsed Nuclear Magnetic Resonance Spectrometer. The need for the fourier transformation of pulsed NMR data arises from the relationship between pulsed and continuous wave NMR data which, in some cases, is defined by the fourier transform. (Gutowsky)

1691 Psychology. Reactions to Risk and Stress. An exploration of the effects of social comparison processes and personality and attitudinal variables on risk-taking behaviors. (J. McGrath)

1692 Food Science. Parameter Estimation of Endospores Germination Kinetics. Calculates the parameters associated with the germination of endospores. These parameters include the maximum rate of germination, time at which it occurs and the fraction of the population of endospores that fail to germinate. The problem is based on the kinetics established by McKormic where an asymptotic regression analysis is made based on an equation derived from a theoretical Weibull distribution. (Mitchell Sogin)

1693 Agricultural Engineering. Optimizing Hydrostatic Transmission Systems. The IBM 360/75 will be used to simulate an engine-hydrostatic transmission system. An attempt will be made to)optimize the system using performance and maximum efficiency criteria, and determine the feasibility of an automatic control system. (F. G. Kostrub)

1694 Geology. Oceanographic Data Reduction. Marine geology data will be reduced and graphically plotted. (A. F. Richards)

1695 T Civil Engineering. Lime-Soda Softening Dosages. Solutions of the simultaneous chemical equilibrium equations involved in calculating the amounts of lime and/or soda ash required for softening of hard waters. (Vogel)

1696 T Theoretical and Applied Mechanics. Dynamics of a Particle on a Cantilever Beam. Numerical solution of the displacement-time function of a particle attached to the end of a cantilever beam for large deflections. The problem solution uses a method of successive approximations to arrive at a point by point solution. (Raymond A. Willem)

1697 T Chemistry and Chemical Engineering. One-Center Hartree-Fock Calculations. The molecular Hartree-Fock equations are solved using molecular orbitals expanded at one nucleus in a series of numerical radial functions and spherical harmonics. This results in a set of coupled 2nd order differential equations, which are integrated numerically using the Numerou method. The wave function eigenvalues are determined iteratively. (R. L. Belford)

1698 Horticulture. Dixon Springs Turf Nutrition Study. The purpose of the research problem is to compare soil fertility programs as affecting plant nutrient, population, and root depth. Basic statistics programs are to be used to find correlations and multiple regression coefficients. The data was obtained in trial plots at Dixon Springs over a period of six years. (Butler)

1687 T Coordinated Science Laboratory. Cognitive Heuristic Problem Solver. The thesis explores the problem of writing heuristic programs whose rules of thumb are adopted and improved by the program itself, based on the recognition of relevant problem features, possibly aided by interaction with a human operator. The problem to be solved by the program is the combinational layout of connected 2-dimensional objects, with minimum connection cost as objective. (Bielowski)

1689 Materials Research Laboratory. Electro-Absorption. An investigation of the band structure of semiconductors from their optical properties. (Paul Handler)

1690 Chemistry and Chemical Engineering. Fourier Transform of Pulsed NMR Spectra. This computation time will be used to calculate, and plot the fourier transform of a function determined by the data output of a Pulsed Nuclear Magnetic Resonance Spectrometer. The need for the fourier transformation of pulsed NMR data arises from the relationship between pulsed and continuous wave NMR data which, in some cases, is defined by the fourier transform. (Gutowsky)

1691 Psychology. Reactions to Risk and Stress. An exploration of the effects of social comparison processes and personality and attitudinal variables on risk-taking behaviors. (J. McGrath)

1692 Food Science. Parameter Estimation of Endospores Germination Kinetics. Calculates the parameters associated with the germination of endospores. These parameters include the maximum rate of germination, time at which it occurs and the fraction of the population of endospores that fail to germinate. The problem is based on the kinetics established by McKormic where an asymptotic regression analysis is made based on an equation derived from a theoretical Weibull distribution. (Mitchell Sogin)

1693 Agricultural Engineering. Optimizing Hydrostatic Transmission Systems. The IBM 360/75 will be used to simulate an engine-hydrostatic transmission system. An attempt will be made to)optimize the system using performance and maximum efficiency criteria, and determine the feasibility of an automatic control system. (F. G. Kostrub)

1694 Geology. Oceanographic Data Reduction. Marine geology data will be reduced and graphically plotted. (A. F. Richards)

1695 T Civil Engineering. Lime-Soda Softening Dosages. Solutions of the simultaneous chemical equilibrium equations involved in calculating the amounts of lime and/or soda ash required for softening of hard waters. (Vogel)

1696 T Theoretical and Applied Mechanics. Dynamics of a Particle on a Cantilever Beam. Numerical solution of the displacement-time function of a particle attached to the end of a cantilever beam for large deflections. The problem solution uses a method of successive approximations to arrive at a point by point solution. (Raymond A. Willem)

1697 T Chemistry and Chemical Engineering. One-Center Hartree-Fock Calculations. The molecular Hartree-Fock equations are solved using molecular orbitals expanded at one nucleus in a series of numerical radial functions and spherical harmonics. This results in a set of coupled 2nd order differential equations, which are integrated numerically using the Numerou method. The wave function eigenvalues are determined iteratively. (R. L. Belford)

1698 Horticulture. Dixon Springs Turf Nutrition Study. The purpose of the research problem is to compare soil fertility programs as affecting plant nutrient, population, and root depth. Basic statistics programs are to be used to find correlations and multiple regression coefficients. The data was obtained in trial plots at Dixon Springs over a period of six years. (Butler)

1699 Electrical Engineering. Effect of Sound on Inner Ear. Mean value calculations and graphical analysis of the effects of high intensity auditory stimuli on the inner ear epitheluim of experimental animals. (H. W. Ades)

1700 Electrical Engineering. Solution of an Open Region Problem as a Limiting Case of a Closed Region Problem. The application of the solution of the bifucated rectangular waveguide to the problem of an open ended waveguide radiating into space will be investigated. This is to be achieved by approaching the limit as the transverse dimension of the large guide becomes indefinitely large. (James Richardson)

1701 Computer Science. Resolution in Automated Theorem Proving. This time will be used to compare methods of choosing sets of support in automated theorem proving by resolution. (L. J. Henschen)

1702 Electrical Engineering. Housekeeping Routines. Certain problems of a utilitarian nature arise from time to time which require access to the services and facilities of DCS. (Roger A. Vossler)

1703 State Geological Survey. Contouring and Trend Surface Analysis. To develop a contouring program for plotting and contouring geological and geophysical data. Trend surface analysis will be used in evaluating the grid point values. (M. Heidari)

1705 Agricultural Engineering. Efficiencies of Hydrostatic Pump. The thesis research involves measuring volumetric, torque, and overall efficiencies at various speeds, pressures, and displacements. The results will be statistically analyzed to determine significant relationships. (Thomas Hanna)

1706 T Agricultural Economics. An Economic Planning Model for Thailand Agriculture. A problem of allocation of resources to projects or activities so as to maximize agricultural production. (Tongro J. Onchandra)

1707 T Electrical Engineering. Numerical Solutions to Differential Game Problems. The problem is concerned with working examples of stocastic and deterministic differential games. (Patrick Shea)

1708 T Agricultural Economics. Financial Optima on Illinois Farms. Linear programming: maximizing a linear function subject to linear inequalities, with use of the simplex algorithm. (Mayberry Baker)

1709 T Nuclear Engineering. Study of the Internal Conversion Electrons in Fission. This investigation will study the internal conversion electrons emitted from fission fragments in the fission of U-235 and Cf-252. Internal conversion electrons are emitted as the result of the de-excitation of low energy excited levels of the fission fragments. Characteristic x-rays are emitted as the vacancies caused by internal conversion are filled by electrons from higher shells. The energies of the these x-rays are characteristic of the charge of the fragment and hence the internal conversion spectra for each atomic charge of the fission fragments can be found. The energies of the electrons and of the x-rays are placed on magnetic tape by a two parameter data acquisition system. The computer is then used to sort the data, and for data analysis such as correction for backscattering and solid angles, and unfolding of spectra. (N. Shapiro)

1710 Veterinary Physiology and Pharmacology. Potassium Diffusion in Tissue. Exchange of potassium between blood and tissue will be studied with the Continuous System Modeling Program. Impedance of diffusion barriers will be determined by best fit of model parameters with experimentally measured extractions and tissue transit times of potassium. (Fred Downey)

1711 Civil Engineering. Prediction of Earthquake Motions. Digital simulation techniques are being employed to aid in the prediction of the effects of earthquake motions on structures. (L. Lopez)

1712 T Sociology. Engagement in Political Affairs According to Family Life Cycle Status. Secondary analysis of data collected by the Survey Research Center for their 1966 election study, to determine if there is a relation between family life cycle status and engagement in political affairs. Three measures of "engagement" will be used: interest, information, and participation. (Donald Conover)

1713 T Materials Research Laboratory. AC Specific Heat. To calculate specific heats from raw data and to fit the resulting values to algebraic expressions. (Garnier)

1714 Aeronautical and Astronautical Engineering. Flight Vehicle Synthesis. Development of flight vehicle simulation and optimization programs.

1715 Psychology. Mathematical Models in Social Psychology. Data analyses from a series of studies of individual and small group problem solving and decision making will constitute the major portion of the computations. Standard analyses of variance, calculation of contingencies among classifications of frequencies, etc., will be common analyses. However, the fitting of probability density functions to data, the calculation of "empirical" convolutions of density functions, the manipulation of matrices are also likely examples of the use of machine time. (James A. Davis)

1716 T Astronomy. Spectroscopic Luminosity Criteria. Data from stellar spectra exist on punched cards. This data will be reduced in the computer and luminosity indices will be calibrated. The luminosities (and thus distances) of a group of program stars can then be determined. (Thomas E. Lutz)

1717 T Chemistry and Chemical Engineering. Extended Huckel MO Calculations of Various Benzene and Methylbenzene Transition Metal Complexes. We are investigating the mechanisms of electron delocalization in transition metal benzene complexes in order to interpret nuclear magnetic resonance data. This requires an extended Huckel calculation of reasonably accurate wave-functions so that hyperfine coupling constants can be evaluated and the contributions from sigma and pi orbitals ascertained. This work is an extension and check of work done by M. F. Rettig on cyclopentadienyl metal systems. We are requesting disk space to save time on these very lengthy calculations. (R. Drago)

1718 Civil Engineering. Photogrammetric Image Coordinate Refinement. The Image Coordinate Refinement phase of Analytical Aerotriangulation corrects measured photo coordinates for atmospheric refraction, lens distortion, film distortion and other sources of image perturbation. This investigation will analyze and compare several methods which are utilized to perform this refinement, and will analyze the sensitivity on final adjusted ground coordinates of variations in the consideration for Image Coordinate Refinement. (G. W. Marks)

1719 T Aeronautical and Astronautical Engineering. Low Thrust Trajectory Optimization. Utilization of TOPCAT I: (Trajectory Optimization for Comparing Advanced Technologies) for mission analysis studies employing low thrust propulsion devices. These studies aim at better classifying advanced state-of-the-art systems with respect to suitable space applications. (Ken Atkins)

1720 Materials Research Laboratory. Diffusion Problem. Change a half hour of 7094 time on problem specification 5Ø019 to 360 time. (Lieberman)

1721 Recreation. Social and Psychological Correlates of Leisure. Multivariate analyses of social and psychological characteristics associated with various forms of leisure activity. Data is obtained primarily from large-scale surveys of different communities. Analyses will be done using FORTRAN programs. (Doyle Bishop)

1722 Zoology. Selection for Dental Traits in the House Mouse. Artificial selection is being applied to the width of the upper first molars of the house mouse, Mus musculus. A high, low, and control line are being maintained. Correlated responses of other dental traits, body weights, and fitness components are being measured. (M. Joan Larson)

1723 Industrial Administration. Team Decision Problems. Stochastic linear programming of management control model of organizational activities. (H. Hinomoto)

1724 Graduate School of Business Administration. Analysis of Alumni Questionnaire. To analyze and evaluate alumni responses to a questionnaire prepared for the College of Commerce and Business Administration. (Rowland)

1725 T Electrical Engineering. Optical Properties of Dielectric Materials. Data reduction including repeated solution of algebraic equations for experiments involving far infrared dielectric materials. (Paul Coleman)

1726 Microbiology. Sedimentation Equilibrium Calculation. The problem involves the calculation of molecular weight from the X and Y coordinates of concentration fringes on interference pictures of high speed sedimentation equilibirium ultracentrifugation studies. It also includes a plot of LNJ vs. R^2. (L. Leon Campbell)

1727 T Education. Study of Attitudes and Approaches to Writing. The study seeks to ascertain whether there are significant differences between selected attitudes and approaches to composition between writers of superior performance and average performance. (Lacampagne)

1728 T Materials Research Laboratory. Analysis of Critical Current Data for Type I Superconductors. The critical current of type I superconductors near the transistion temperature will be investigated. The experimental data will be analyzed and compared with London's theory and flux flow theories. The experimental parameters are the superconducting metal; its diameter, its temperature, and its surface condition which includes normal metals plated on the surface. (Ival Toepke)

1729 Electrical Engineering. Charged Droplet-Gas Interactions. The fields surrounding a highly charged liquid droplet may be sufficient to ionize the surrounding gas and/or modify the droplet evaporation rate. This problem pertains to the calculation of fields and diffusive flow around such a drop. (John Robertson)

1730 Materials Research Laboratory. Listings. Inventory listings will be made on the model 20. (Roy King)

1731 Agricultural Economics. Economics of Cattle Feeding. Use of MPS-360 routine - linear programming. (VanArsdall)

1732 T Electrical Engineering. Control Systems. Optimal control of discrete systems with random parameters. (Valleni)

1733 Chemistry and Chemical Engineering. Dynamic Structure of Fluids. Development and testing of mathematical models for random motions in liquids; calculations of correlation times for molecular motion from nuclear magnetic resonance relaxation measurements. (J. Jonas)

1734 Vocational and Technical Education. Practical Nursing Study. A demographic study of practical nurses and practical nursing students including their vocational behavior and occupational mobility. (Tomlinson)

1736 Marketing. Purchasing Behavior Predicted by N-Ach. The relationship between levels of need for achievement and present and desired inventory of goods and services. (David M. Gardner)

1737 T Mechanical Engineering. Forced Convection in Laminar Boundary Layer. Forced convection laminar boundary layer over wedges of arbitrary temperature distribution. An analytic approach has been followed. (L. S. Cheema)

1738 T Electrical Engineering. Random Convection Networks. Determination of connection probabilities among the sites of two and three dimensional lattices as a function of lattice size and site to site connection probabilities. (Lester Lendrum)

1739 Materials Research Laboratory. Many Body Effects on Lattice Vacancies. The energy of formation of lattice vacancies will be calculated considering many body forces as well as two body forces. The atoms around the vacancy will be allowed to relax to a minimum energy configuration. The minimum energy configuration will be found using Newton Raphson technique. (James Burton)

1741 T Mining, Metallurgy, and Petroleum Engineering. River Meanders. A statistical approach to the problem of river meanders is contemplated. Spectral analysis of the observed data and statistically generated data is to be made. (Thakur Tukrel)

1742 Microbiology. Enzyme Kinetics. Renewal of problem specification 1009. (R. D. Demoss)

1743 T Zoology. Geographic Variation and Systematics of the Plethodonted Genus Pseudoeurycea. Geographic variation within the Pseudoeurycea belli-Plethodontidae species group using statistical methods to test significance. Of chief concern is the degree of difference between populations as indicated by meristic and color pattern characters. Meristic characters are being compared using an analysis of variance of regression lines. Pattern characters are converted into frequency distribution and these distributions are then compared using chi square analysis. (Landy Macreay)

1744 T Materials Research Laboratory. Calculating Energy Levels Observed Experimentally by Microwave Spectroscopy. Calculation of energy levels as described will require generating and diagonalizing several matrices. Several parameters will have to be fit to the data, and a series of matrices will have to be generated and diagonalized to fit each one. The experimental data have been obtained already. (R. S. Scott)

1745 T Civil Engineering. Short Range Photogrammetry. This study involves the adaptation of photogrammetric techniques for use in Civil Engineering laboratory projects such as static and dynamic testing of structural models. (R. C. Malhotra)

1746 T Electrical Engineering. Numerical Analysis of Aperture Antennas. The numerical solution of the integral equations associated with aperture antennas will be attempted. Investigation of cylindrical dipole using this technique will be extended. (A. J. Poggio)

1747 Psychology. Infant Response to Novelty and Familiarity. Infants above eights weeks of age show more interest in novel than in familiar objects. We are interested, however, in whether they go through a prior stage of attachment to familiar objects. Such a finding would have many implications for theories of cognitive development. We have tested the relative preferences for novel or familiar objects (using visual fixation time as a measure) of a group of infants at six and again at eight weeks of age. We have predicted a shift from preference for the familiar to preference for the novel with increasing age. A second group of infants was tested only once, at eight weeks, to control for the effects of the six weeks testing experience itself. (Weizmann)

1748 Materials Research Laboratory. Surface-Induced Infrared Absorption. The surface induced absorption in alkali halide crystals is to be calculated using Green's function techniques. (T. P. Martin)

1749 Illini Christian Fellowship. Student Organization Mailing List Organization. The Illini Christian Fellowship desires the use of the computing facilities for the automatic processing of address lists: sorting and listing by various categories, and automatic production of name and address labels.

1751 Nuclear Engineering. Particle Trajectory and Turbulent Dispersion. Statistical data analysis will be performed on turbulent trajectory data of suspended particles. Dispersion will be evaluated from data and from analytical predictions. The research is related to sedimentation and pollution in streams and atmospheric turbulence. (B. G. Jones)

1752 T Mechanical Engineering. Electrohydrodynamic Secondary Flow. The solution of poissons type of equations which describe the electric field distribution between the two parallel conducting planes of the electrostatic precipitator, the results obtained are to be used in the solution of the general equations for the electric wind secondary flow. (O. E. Ramadan)

1753 Electrical Engineering. Imprisonment of Resonant Radiation. Research in gaseous electronics dealing with solutions to various mathematical problems. (J. T. Verdeyen)

1755 T Civil Engineering. Stress Waves in Elastic Media. For certain problems of dynamic contact between rigid dies and an elastic half-space, the state of stress at any point within the elastic medium may be written in the form of an hyper-elliptic integral. The variable of integration in such a problem is complex. A program has been written to evaluate these integrals numerically. (J. C. Thompson)

1756 T Nuclear Engineering. Calculation of Probability Distributions for Photomultipliers. Calculate electron probability distributions for electrons in a dynamic crossed field photomultiplier. These distributions will relate to electron impact energy, impact position, and time of impact. (Donald J. Leverenz)

1757 T Materials Research Laboratory. Thermal Conductivity of Vanadium. Experimental data will be used to calculate the thermal conductivity of superconducting Vanadium as a function of the applied external magnetic field and the temperature of the sample. The temperature will be varied from one degree absolute to the transition temperature around five degrees. The calculation will involve curve fitting to calibration data for both carbon and Germanium thermometers and determination of the slope of these curves. (B. Keith Moore)

1758 T Food Science. Radiochromatography of $[H^3]$ Hexitols Using $[C^{14}]$ Glucose as Standard. Quantitation of $[H^3]$ hexitols in radiochromatograms by initially subtracting $[C^{14}]$ counts from $[H^3]$ counts, converting $[H^3]$ counts to n moles of hexitols and normalization of the results via the $[C^{14}]$ glucose internal standard. (M. P. Veniamin)

1760 Materials Research Laboratory. Tunneling in Semiconductors. The conductance of a metal-oxide-semiconductor tunnel junction is to be calculated. The effects of the interaction of the tunneling electrons with phonons and plasmons will be studied. (L. C. Davis)

1761 Modine Manufacturing Company. Industrial Salvage of Used Solder Flux. This problem is concerned with an industrial salvage of used solder flux solutions. Given the volume and concentration of a used solution, the problem is to determine the amount of water necessary to bring the solution to a usable concentration. Because the volume and the concentration are independent of each other, the number of combinations which must be considered is 760. The amount of calculations needed indicate that using a computer would be the most efficient and accurate method. (James W. Davis)

1763 Student Financial Aids. Organizing Office Data. Sorting data to obtain various printouts. (Anthony M. Engels)

1766 Natural History Survey. Fish Population Dynamics. To determine the effect of species interactions on recruitment, total production, and yield of various species of freshwater fishes. (William Childers)

1772 Education. Evaluation Undergraduate Education Courses. This research is an attempt to evaluate the extent to which objectives of professional courses in the undergraduate teacher education sequence are achieved. Data were obtained from students enrolled in courses and from supervising teachers in public schools. Frequency counts, means, standard deviations, and differences of means tests will be extracted from data which have been punched on IBM cards. (Lereh-Carver)

1777 Accountancy. Application of Current Market Value Accounting Model. Use of non-linear least squares regression for determination of asset values. (McKeown)

1780 Materials Research Laboratory. Integral Equations for Dilute Boron Mixtures. The solution of coupled integral equations to obtain the ground state energy and pair correlation functions for dilute solutions of quantum fluids. (Walter Massey)

1781 Theoretical and Applied Mechanics. Theoretical and Applied Mechanics Department Research. This number is to be used for individual research in connection with teaching TAM 223 and 224. Analysis of data for evaluating material behavior. (H. R. Wetenkamp)

1782 T Theoretical and Applied Mechanics. Scatterer Light Photo-elasticity. The problem is to develop a method for determining stress distributions by means of scattered light photoelasticity. This project is to improve on experimental procedures and to incorporate a more systematic means of data reduction. (Martin)

1783 Physics. Composite Systems. These programs are part of an investigation of the composite structure of elementary particles as evidenced by scattering experiments. (James Trefil)

1786 Mechanical Engineering. Elasto-Hydrodynamic Gear Lubrication. This investigation will be concerned with heavily loaded gear lubrication. The effects of tooth deformation, both Hertz and beam deflection, will be considered and the viscosity will be considered a function of the pressure in the oil film. The problem is non-linear since the tooth deformation depends upon the film pressure which in turn depends upon the tooth shape and viscosity. Numerical integration procedures will be used to obtain the film pressures and deformed shape of the teeth. Interation procedures will be developed to satisfy boundary conditions. (R. W. Adkins)

1788 Education. Education Graduate Advising System. We propose to design and implement a system of integrated computer programs which can be used to create and maintain a data base consisting of records of the 1,000 graduate students in the College of Education. The data base will be constructed and the programs will be tested during the fall semester of 1968; after that, the student file will be updated each school term. (Carss)

1790 Animal Science. Maternal Diet and Management of Swine. To study the effects of maternal diets on the reproduction of the adult pig and subsequent performance of offspring. (B. G. Harmon)

1795 Materials Research Laboratory. Analytical Spectroscopic Determinations. Evaluation of analytical data. (J. A. Eakin)

1796 Mechanical Engineering. Supersonic Flow Field Calculation. Transforming our 7094 programs to 360 (the programs developed for the calculations of symmetric supersonic flow field). (W. L. Chow)

1797 T Political Science. Computer Simulation of the Legislative Process. Thesis research including the development, testing and experimentation on a computer simulation of legislative behavior. (L. V. Grant)

1802 Accountancy. Value of Information Analysis. This is a non-linear program simulation of the value of information for small versus large firms in the United States. It will involve a series of partial problems and analysis of them and one final comprehensive problem drawing the separate studies together. (Norton Bedford)

1803 T Zoology. Phylogenetic Relationships of the Rhacophorids Frogs. Using cluster analysis for deriving phylogeny of the rhacophorids frogs. The program and analysis is based on Sharrook's method (not published - University of Chicago). This method currently being used at the Field Museum of Natural History, Chicago. (Sioe Liem)

1804 Civil Engineering. Finite Element Analysis of Axially Symmetric Solids of Revolution. The program is to be developed for the stress analysis of axially symmetric solids of revolution for various loading or boundary conditions by the finite element method. The material of the structure can be in the elasticplastic range. Complex bodies of many different materials are to be represented. The program has to work for both constant stress and constant strain through an element. (Saad Rashid)

1805 T German. Analysis of Gothic Bible. A program is being written to remove from magnetic tape certain data taken from a Gothic version of the Bible. The data will be analyzed to substantiate work done for a dissertation. (Viehmeyer)

1806 T Chemistry and Chemical Engineering. Enclosed Rotating Flow. Finite difference studies will be made on rotating flows in enclosures. (J. L. Hudson)

1809 T Chemistry and Chemical Engineering. Correlated Atomic Orbital Calculations. Numerical determination of correlated atomic orbitals for the ground- and low-lying excited states of helium-like ions and the beryllium atom through the use of first-order perturbation theory. (Daniel Davis)

1815 Institute of Communications Research. Individual Differences in Human Judgment. The problem entails a series of standard statistical analyses designed to determine individual differences in human judgment. (N. Wiggins)

1819 T Theoretical and Applied Mechanics. Concrete Columns Reinforced with High Strength Steels Subjected to Biaxially Eccentric Loading. A theoretical and experimental analysis of the behavior, ultimate strength and mode of failure of biaxially eccentrically loaded rectangular concrete columns with corner reinforcement of high strength steels. (P. Heimdahl)

1820 T Vocational and Technical Education. A Comparison of Two Concurrent Work Education Models. A comparison of two models for obtaining concurrent work education in agriculture. (Bobbitt)

1823 Electrical Engineering. Coalescence of Drops in Liquids. Investigation of the interaction of drops of liquid in another immiscible liquid. The growth of drops by coalescence in the presence of an electric field will be studied. (Hendricks)

1827 State Water Survey. Sub-Synoptic Circulations and Their Effect on Radioactive Material. A diagnostic study of sub-synoptic local circulations in the upper and middle troposphere is being made in order to improve estimates of the rate of diffusion of radioactive material under different conditions of large-scale flow. (John Wilson)

1828 Materials Research Laboratory. Thermal Conductivity Curve Fitting. The purpose of this project is to calculate curves of the thermal conductivity versus temperature for the transition-metal carbides. (Lee Radosevich)

1836 T Electrical Engineering. An Analytical and Computer Study of a Linear Actuator. A study of a linear actuator using the Continuous System Modeling Program to determine a transfer function. (John Erickson)

1842 Mechanical Engineering. Duct Optimization. Optimization of an air distribution for buildings using method of Lagrange multipliers. (Winn)

1844 Office of Research in Medical Education. Opthopaedic Training Study. Study of the effects of training conditions on performance in Orthopaedic residency training. (Thomas Bligh)

1845 T Computer Science. Study of High-Speed Digital Division. The purpose of this research is to perform cost-effectiveness studies of high-speed digital division techniques. Of particular interest will be new variations which are now feasible due to the relatively low per gate cost and high packaging density afforded by integrated circuits and large-scale integration. Also of special interest is a comparison of linear techniques which generate quotient digits at a fixed rate, and quadratic techniques which double the number of quotient digits generated on each

iteration. The languages to be used are FORTRAN IV, PL/I, and the General Purpose Simulation System (GPSS). (D. E. Atkins)

1847 Nuclear Engineering. Deterministic Monte Carlo. The problem proposed is to develop Monte Carlo codes which handle each phase of a particle history in a predetermined number of steps. Such codes are required if many histories are to be performed simultaneously under the direction of a single sequence of instructions. In the Monte Carlo calculations currently performed the number of steps per history is variable. One reason for this is that the selection of numbers randomly distributed according to some probability distribution function will occur at some phase of a particle history. This selection frequently involves use of a rejection technique. The number of steps required to complete the selection by this method is not fixed. Hence, one important aspect of the problem is to develop sampling techniques whereby selection of numbers from a probability distribution can be performed in a fixed number of steps. (Theron Carlson)

1848 T Mechanical Engineering. Study of Instantaneous Load on Gear Teeth. A study will be made of the instantaneous loads to which spur gear teeth are subjected during normal operating conditions. The study will include effects due to mass and elasticity in the system of meshing gears. (Wm. F. Hahn)

1851 Mechanical Engineering. Temperature Recovery. Transient behavior of building and heating system. (Turek)

1855 T Physics. Lattice Statistics. Use of statistical mechanical series expansions to study the properties of lattices. (Martin Ferer)

1856 Electrical Engineering. SSP Deck Dump. The purpose of this problem specification number is to obtain copies of various SSP decks. (Roger A. Vossler)

1858 Engineering Division. Engineering Research. Research of Public Works design and applications. (Stephen H. Schaefer)

1861 T Computer Science. Wiener Integral for 3rd Virial Coefficient. Direct and exchange terms for the 3rd virial coefficient for dilute He^4 gas at temperatures below $5^{\circ}K$ are computed by evaluating functional integrals using Monte-Carlo technique. (R. Jacobson)

1863 T Chemistry and Chemical Engineering. High Temperature Shock Tube Kinetics Calculations. A shock tube fitted with mass spectral and photometric detection systems is being used to make high temperature gas phase reaction kinetics studies. The computer is needed to perform the shock calculations and to reduce the raw kinetics data to its most meaningful form. (Richard J. D'Amato)

1866 T Atronomy. Solution of Euler's Equations in the Linear Wave Approximation. The problem is to determine the changes in size and shape of a homogeneous ellipsoid of gas whose motion is governed by the linear wave approximation for the velocity, the internal gravitational potential and an adiabatic pressure gradient. With these assumptions, the problem reduces to the solution of from one to fifteen ordinary differential equations; the solutions of which are to be obtained numerically. (Peter Melvin)

1867 Mechanical Engineering. Absorption System Simulation. System simulation of absorption refrigeration units. (Hulsbrink)

1870 Pershing Rifles. Use of 360/20. Use of printer for printing out mailing lists. (Cich)

1880 T Civil Engineering. Stresses in Rail. This project involves the study of stresses in a rail head. (Martin)

1881 Civil Engineering. Simulation of Ballast Support. This project involves the development of a mathematical model of railroad ballast support. (Lundgren)

1882 Civil Engineering. Evaluation of Rail Sections. This project involves a mechanical and economical analysis of railroad rail sections. (Butler)

1884 University Office of School and College Relations. Administrative Research on Admission Procedures. Use of 360 to analyze research of administrative admission procedures. (L. F. Robinson)

1891 T Agronomy. Mass Spectrograph Computations. This program is designed to compute mass peak heights, determine mass 29/mass 28 ratios, calculate atom $\%N^{15}$ and to calculate and correct for any air N contamination. (Westerman)

1893 T Materials Research Laboratory. Analysis of Tracer Diffusion Data. Tracer diffusion data, obtained by lathe sectioning and standard counting methods, are to be analyzed. This will include least-squares fitting the data to a straight line on a semi-logarithmic plot and using various PLOT subroutines to display the results. Programs currently written for the 7094 are to be converted over to the 360. (Rondo N. Jeffery)

1894 Atronomy. Processing Stellar Photographic Data. Photographic plates of stellar objects are processed in the microphotometer. The digitized data (plate transmission and plate position) are recorded on magnetic tape. The tape is then converted to useful information via the computer. (Kenneth Yoss)

1896 T Coordinated Science Laboratory. Simulation of an Associative Processor. The Associative Processor (AP) is a linear array of cells containing a fixed length word (~40 bits) with logic in each cell capable of detecting a match in a search operation. The 360/75 will be used to simulate one or more models of an A.P. and the simulation will include I/O communication of the A.P. using the I/O of the 360/75. (Lipovski)

1898 Chemistry and Chemical Engineering. Theoretical Inorganic Investigations. Semi-empirical molecular orbital calculations on a wide variety of inorganic molecules will be undertaken. (Drago)

1904 Mechanical Engineering. Analysis of Cavities with Self-Induced Pressure Oscillations. Cut-outs or notches which appear in the surface of aerodynamic vehicles often generate noise in the external stream due to self-induced pressure oscillations within the cavity. These pressure variations cause increases in both the local aerodynamic heating and the drag of the body. Wind tunnel data obtained in the universities facilities will be reduced and used to provide insight into the physical processes occurring within the cavity. Once the governing problems are determined they will be solved using standard numerical techniques. (Robert A. White)

8.4 Class Problem Specifications

During the third quarter of 1968, 109 problem specifications were submitted to cover all assigned problems on the System/360 in the following courses.

1659	Mechanical Engineering 264.
1662	Theoretical and Applied Mechanics 493.
1664	Nuclear Engineering 451.
1665	Civil Engineering 368.
1666	Political Science 450.
1671	Music 320.
1672	Mechanical Engineering 429.
1676	Agricultural Economics 200.
1677	Division of University Extension 999.
1688	Civil Engineering 201.
1704	Psychology 492.
1735	Plant Pathology 302.
1740	Social Work 491.
1750	Agricultural Engineering 499.
1754	Agronomy 365.

1759 Mechanical Engineering 221.

1762 Aeronautical and Astronautical Engineering 326.

1764 Mechanical Engineering 271.

1765 Architecture 456.

1767 Agricultural Engineering 286.

1768 Computer Science 201.

1769 Computer Science 209.

1770 Computer Science 290.

1771 Chemistry and Chemical Engineering 398.

1773 Aeronautical and Astronautical Engineering 264.

1774 Aeronautical and Astronautical Engineering 241.

1775 Chemistry and Chemical Engineering 466.

1776 Civil Engineering 497.

1778 Computer Science 101.

1779 Computer Science 101.

1784 Geology 479.

1785 Mechanical Engineering 448.

1787 Mechanical Engineering 341.

1789 Computer Science 103.

1791 Physics 303.

1792 Electrical Engineering 250.

1793 Nuclear Engineering 401.

1794 Chemistry and Chemical Engineering 392.

1798 Theoretical and Applied Mechanics 416.

1799 Computer Science 400.

1800 Civil Engineering 391.

1801 Civil Engineering 262.

1807 Mechanical Engineering 264.

1808 Mechanical Engineering 263.

1810 Civil Engineering 297.

1811 Computer Science 491.

1812 Agricultural Engineering 311.

1813 Engineering Honors 297.

1814 Business Administration 373.

1816 Zoology 303.

1817 Electrical Engineering 251.

1818 Computer Science 301.

1821 Theoretical and Applied Mechanics 235.

1822 Chemistry and Chemical Engineering 348.

1824 Electrical Engineering 330.

1825 Computer Science 101.

1826 Electrical Engineering 323.

1829 Mechanical Engineering 185 and 186.

1830 Nuclear Engineering 455.

1831 Mechanical Engineering 421.

1832 General Engineering 288.

1833 Civil Engineering 361.

1834 Chemistry and Chemical Engineering 421.

1835 Electrical Engineering 423.

1837 Architecture 494.

1838 Music 304.

1839 Business Administration 545.

1840 Business Administration 532.

1841	Agricultural Economics 325.
1843	Political Science 430.
1846	Mechanical Engineering 296.
1849	Agricultural Engineering 276.
1850	Nuclear Engineering 451.
1852	Mechanical Engineering 293.
1853	Mechanical Engineering 406.
1854	General Engineering 103.
1857	Geology 493.
1859	Computer Science 457.
1860	Educational Psychology 490.
1862	Chemistry and Chemical Engineering 367.
1864	Electrical Engineering 497.
1865	Industrial Engineering 282.
1868	Business Administration 560.
1869	Electrical Engineering 498.
1871	Mechanical Engineering 306.
1872	Civil Engineering 369.

1873	Theoretical and Applied Mechanics 493.
1874	Theoretical and Applied Mechanics 293.
1875	Civil Engineering 264.
1876	Electrical Engineering 342.
1877	Electrical Engineering 389.
1878	Mechanical Engineering 205.
1879	Civil Engineering 250.
1883	Computer Science 387.
1885	Civil Engineering 335.
1886	Civil Engineering 230.
1887	Civil Engineering 497.
1888	Civil Engineering 306.
1889	Educational Psychology 390.
1890	Electrical Engineering 350.
1892	Computer Science 999.
1895	Theoretical and Applied Mechanics 311.
1897	Electrical Engineering 322.
1899	Mechanical Engineering 256.

1900 Mechanical Engineering 293.

1901 Theoretical and Applied Mechanics 428.

1902 General Engineering 293.

1903 Aeronautical and Astronautical Engineering 414.

1905 Chemistry and Chemical Engineering 343.

9. IBM 7094/1401 SERVICE, USE, AND DEVELOPMENT

(Supported in part by the National Science Foundation under Grant No. NSF-GP-700.)

9.1 Log Summaries

Table I - IBM 1401-II

Summary of Use

July, 1968

Scheduled Engineering	6:00
Unscheduled Engineering	14:05
Maintenance	11:17
7094 Preparation	540:39
List/Reproduce	49:58
Code Check	1:38
Tape Dump	6:30
1604 Preparation	:15
Idle	30:06
Total	660:28

Table II - IBM 1401-II

Summary of Machine Errors

July, 1968

1402 Card Reader Punch	4
1403 Printer	2
729V Tape Drives	1
Total	7

Table I - IBM 7094

Summary of Use

July, 1968

Scheduled Engineering			24:59
Unscheduled Engineering			12:07
Maintenance			4:27
Idle			4:26
Miscellaneous (Operator training, tape rewind, system tape mounting, rerun of failing problems)			89:23
Total Use			
Training and Education		6:02	
University Administrative Overhead Use		5:26	
System Modification and Improvement		21:58	
System Accounting		7:39	
System Updating		:29	
Customer Use			
In System	433:39		
Special Short Shots	1:11		
Customer Use Total		434:50	
	Total Use		476:24
	Total Time		611:46

Table II - IBM 7094

Summary of Errors

July, 1968

Disk		2
	Total	2

IBM 7094 TABLE III JULY, 1968

DEPT[1]	NUMBER OF RUNS			NUMBER OF SPECS			IBM 7094 USAGE IN HOURS-MINUTES-SECONDS			
	TANDE[2]	RES	TOTAL	TANDE[2]	RES	TOTAL	TANDE[2]	RESEARCH	REFUND	TOTAL
AAE	0	48	48	0	2	2		4 16 36		4 16 36
ACCY	0	41	41	0	2	2		6 10		6 10
ADV	0	59	59	0	6	6		3 33 37		3 33 37
AGE	0	180	180	0	6	6		2 40 14		2 40 14
AGEC	0	309	309	0	13	13		5 28 03		5 28 03
AGRON	0	61	61	0	9	9		3 08 39		3 08 39
ANS	0	68	68	0	4	4		1 27 24		1 27 24
ANTH	21	15	36	1	1	2	58 22	20 45		1 19 07
ARCH	3	0	3	1	0	1	32			32
ASTR	0	181	181	0	5	5		3 38 30		3 38 30
ASTRMI	0	20	20	0	1	1		39 00		39 00
BINRES	0	40	40	0	2	2		40 04		40 04
BOT	0	23	23	0	2	2		14 23		14 23
CCSCS	0	14	14	0	1	1		11 45		11 45
CE	76	326	402	1	22	23	38 06	4 17 03		4 55 09
CHE	0	1122	1122	0	35	35		27 49 36		27 49 36
CICBIO	0	15	15	0	1	1		8 31		8 31
CP	0	14	14	0	1	1		10 40		10 40
CRC	0	5	5	0	2	2		21 53		21 53
DCS	0	17	17	0	2	2		23 52		23 52
DS	0	7	7	0	2	2		1 58		1 58
ECON	8	129	137	1	5	6	8 56	49 25		58 21
ED	0	118	118	0	11	11		4 23 49		4 23 49
EDPSY	6	2	8	1	1	2	14 52	50		15 42
EE	0	464	464	0	20	20		10 30 49		10 30 49
ENGCST	0	1	1	0	1	1		4 44		4 44
ENTOM	0	30	30	0	3	3		23 09		23 09
FIN	0	26	26	0	1	1		15 43		15 43
FOR	0	10	10	0	2	2		22 07		22 07
GENE	67	70	137	1	2	3	18 00	1 09 00		1 27 00
GEOG	0	30	30	0	1	1		22 26		22 26
GEOL	0	131	131	0	3	3		2 41 16		2 41 16
HEC	0	6	6	0	1	1		15 39		15 39
HED	0	106	106	0	1	1		10 20 27		10 20 27
HONDRS	0	20	20	0	1	1		35 24		35 24

IBM 7094 TABLE III JULY, 1968 -CONT

DEPT[1]	NUMBER OF RUNS TANDE[2]	RES	TOTAL	NUMBER OF SPECS TANDE[2]	RES	TOTAL	IBM 7094 USAGE IN HOURS-MINUTES-SECONDS TANDE[2]	RESEARCH	REFUND	TOTAL
HORT	0	35	35	0	3	3		32 08		32 08
ICR	0	167	167	0	2	2		5 22 01		5 22 01
IED	0	119	119	0	1	1		5 02 16		5 02 16
IGPA	0	3	3	0	1	1		18		18
ILR	0	83	83	0	3	3		57 38		57 38
INADM	0	46	46	0	1	1		57 21		57 21
IREC	0	62	62	0	2	2		2 27 28		2 27 28
LIBS	0	3	3	0	1	1		1 12		1 12
MATH	0	21	21	0	1	1		1 14 34		1 14 34
MATRL	0	1043	1043	0	26	26		39 25 32		39 25 32
MCBIO	0	48	48	0	3	3		20 33		20 33
ME	52	1032	1084	3	21	24	13 47	28 38 43		28 52 30
MMPE	0	157	157	0	1	1		1 55 44		1 55 44
MUSIC	13	101	114	1	1	2	7 26	8 06 43		8 14 09
NHS	0	8	8	0	1	1		24 00		24 00
NUCE	0	513	513	0	11	11		15 20 33		15 20 33
OIR	0	259	259	0	1	1		10 43 01		10 43 01
PEM	0	256	256	0	17	17		1 59 46		1 59 46
PHYB	0	57	57	0	3	3		1 08 16		1 08 16
PHYCS	0	423	423	0	10	10		8 25 31		8 25 31
PHYSL	0	15	15	0	2	2		16 40		16 40
PHYX	0	1732	1732	0	10	10		124 13 48		124 13 48
POLS	65	352	417	2	5	7	3 22 14	15 57 09		19 19 23
PSYCH	0	1251	1251	0	34	34		24 43 58		24 43 58
REC	0	56	56	0	3	3		6 30 03		6 30 03
SCONS	0	68	68	0	1	1		31 22		31 22
SCS	0	11	11	0	1	1		6 00		6 00
SGS	0	25	25	0	2	2		1 53 45		1 53 45
SOC	0	185	185	0	6	6		7 23 25		7 23 25
SOCW	0	19	19	0	1	1		23 38		23 38
SOLS	0	1	1	0	1	1		36		36
SPCH	0	20	20	0	2	2		7 00		7 00
SPED	0	9	9	0	1	1		14 42		14 42
SRL	0	143	143	0	2	2		17 55 50		17 55 50
SWS	0	261	261	0	14	14		6 23 07		6 23 07
TAM	0	149	149	0	7	7		1 29 09		1 29 09

IBM 7094 TABLE III JULY, 1968 -CONT

DEPT[1]	NUMBER OF RUNS			NUMBER OF SPECS			IBM 7094 USAGE IN HOURS-MINUTES-SECONDS			
	TANDE[2]	RES	TOTAL	TANDE[2]	RES	TOTAL	TANDE[2]	RESEARCH	REFUND	TOTAL
VPH	0	2	2	0	1	1		1 33		1 33
VPP	0	7	7	0	1	1		2 06		2 06
VTED	0	1	1	0	1	1		18		18
ZOOL	0	46	46	0	4	4		26 22		26 22
SUBTOTAL	311	12497	12808	12	381	393	6 02 15	433 37 20		439 39 35
DCSSYS[3]	0	667	667	0	10	10		29 36 47		29 36 47
SSUAD[4]	0	88	88	0	1	1		5 25 33		5 25 33
XDCS[5]	0	12	12	0	1	1		29 11		29 11
XSSS[6]	0	176	176	0	1	1		1 11 02		1 11 02
SUBTOTAL	0	943	943	0	13	13		36 42 33		36 42 33
TOTALS	311	13440	13751	12	394	406	6 02 15	470 19 53		476 22 08

1 See list of departmental codes following
2 Training and Education
3 System Modification and Improvement
4 University Administrative Overhead Use
5 System Updating
6 Special Short Shots

Table I - IBM 1401-II

Summary of Use

August, 1968

Scheduled Engineering	5:10
Unscheduled Engineering	17:59
Maintenance	5:43
7094 Preparation	528:48
List/Reproduce	38:34
Code Check	:30
Tape Dump	10:21
1604 Preparation	1:52
ILLIAC II Preparation	:55
Idle	54:51
Total	664:43

Table II - IBM 1401-II

Summary of Machine Errors

August, 1968

1402 Card Reader Punch	7
1403 Printer	5
Total	12

Table I - IBM 7094

Summary of Use

August, 1968

Scheduled Engineering			17:54
Maintenance			3:17
Air Conditioning			10:54
Idle			81:32
Miscellaneous (Operator training, tape rewind, system tape mounting, rerun of failing problems)			94:32
Total Use			
Training and Education		14:06	
University Administrative Overhead Use		7:32	
System Modification and Improvement		16:48	
System Accounting		10:24	
System Updating		:14	
Customer Use			
In System	365:08		
Special Short Shots	:26		
Customer Use Total		365:34	
	Total Use		414:38
	Total Time		622:47

Table II - IBM 7094

Summary of Errors

August, 1968

There were no errors during August, 1968.

IBM 7094 TABLE III AUGUST, 1968

DEPT[1]	NUMBER OF RUNS TANDE[2]	RES	TOTAL	NUMBER OF SPECS TANDE[2]	RES	TOTAL	IBM 7094 USAGE IN HOURS-MINUTES-SECONDS TANDE[2]	RESEARCH	REFUND	TOTAL
AAE	0	116	116	0	2	2		4 54 24		4 54 24
ACCY	0	65	65	0	4	4		27 42		27 42
ADV	0	47	47	0	4	4		3 05 40		3 05 40
AGE	0	191	191	0	4	4		2 30 08		2 30 08
AGEC	0	204	204	0	8	8		3 48 35		3 48 35
AGRON	0	63	63	0	7	7		1 11 21		1 11 21
ANS	0	83	83	0	5	5		2 45 01		2 45 01
ANTH	11	11	22	1	1	2	1 00 21	44 52		1 45 13
ARCH	3	0	3	1	0	1	4 04			4 04
ASTR	0	107	107	0	5	5		2 41 29		2 41 29
ASTRMI	0	24	24	0	1	1		1 51 32		1 51 32
BINRES	0	86	86	0	2	2		2 00 32		2 00 32
BOT	0	2	2	0	1	1		32		32
CE	13	276	289	1	17	18	3 54	7 30 45		7 34 39
CHE	0	937	937	0	31	31		23 18 16		23 18 16
CICBIO	0	4	4	0	1	1		1 44		1 44
CIRCE	0	7	7	0	1	1		29 16		29 16
CP	0	7	7	0	1	1		8 49		8 49
CRC	0	17	17	0	3	3		20 19		20 19
DS	0	22	22	0	2	2		4 00		4 00
ECON	13	166	179	1	6	7	21 14	1 48 55		2 10 09
ED	0	133	133	0	10	10		3 40 09		3 40 09
EDPSY	5	0	5	1	0	1	5 31			5 31
EE	0	300	300	0	17	17		7 40 16		7 40 16
ENTOM	0	14	14	0	1	1		4 04		4 04
FIN	0	50	50	0	2	2		46 04		46 04
FOR	0	19	19	0	2	2		50 23		50 23
GENE	48	28	76	1	2	3	22 19	13 26		35 45
GEOL	16	43	59	1	3	4	25 33	31 10		56 43
HED	0	33	33	0	1	1		2 12 21		2 12 21
HLTHSV	0	21	21	0	1	1		45 03		45 03
HONORS	0	21	21	0	1	1		45 03		45 03
HORT	0	21	21	0	3	3		19 07		19 07
HOUDIV	0	4	4	0	1	1		10 08		10 08
ICR	0	207	207	0	2	2		5 54 21		5 54 21

IBM 7094 TABLE III AUGUST, 1968 -CONT

DEPT[1]	NUMBER OF RUNS TANDE[2]	RES	TOTAL	NUMBER OF SPECS TANDE[2]	RES	TOTAL	IBM 7094 USAGE IN HOURS-MINUTES-SECONDS TANDE[2]	RESEARCH	REFUND	TOTAL
IED	0	47	47	0	1	1		1 01 48		1 01 48
IGPA	0	49	49	0	1	1		1 41 45		1 41 45
ILR	0	35	35	0	2	2		43 43		43 43
INADM	0	65	65	0	1	1		47 27		47 27
IREC	0	36	36	0	3	3		40 40		40 40
LIBS	0	2	2	0	1	1		50		50
MATH	0	11	11	0	1	1		1 22 33		1 22 33
MATRL	0	1C69	1069	0	24	24		35 08 00		35 08 00
MCBIO	0	63	63	0	3	3		27 24		27 24
ME	5	1C03	1008	2	21	23	43	30 29 37		30 30 20
MMPE	0	105	105	0	1	1		1 52 08		1 52 08
MUSIC	52	46	98	1	1	2	4 27 36	10 03 21		14 30 57
NUCE	0	285	285	0	8	8		22 22 13		22 22 13
OIR	0	156	156	0	1	1		4 36 21		4 36 21
PEM	0	111	111	0	12	12		1 24 44		1 24 44
PHYB	0	139	139	0	2	2		2 41 45		2 41 45
PHYCS	0	407	407	0	7	7		6 05 46		6 05 46
PHYSL	0	10	10	0	2	2		7 11		7 11
PHYX	0	1230	1230	0	10	10		104 48 14		104 48 14
POLS	93	18	111	2	4	6	7 14 20	2 55 21		10 09 41
PSYCH	0	1C84	1084	0	33	33		26 00 36		26 00 36
REC	0	24	24	0	3	3		2 23 12		2 23 12
SCCNS	0	70	70	0	1	1		24 07		24 07
SGS	0	50	50	0	2	2		39 10		39 10
SOC	0	99	99	0	8	8		8 38 10		8 38 10
SOCW	0	7	7	0	1	1		16 01		16 01
SPCH	0	15	15	0	1	1		36 54		36 54
SPED	0	12	12	0	2	2		6 03		6 03
SRL	0	72	72	0	2	2		4 17 05		4 17 05
SWS	0	277	277	0	15	15		7 13 41		7 13 41
TAM	0	79	79	0	6	6		50 00		50 00
VPH	0	2	2	0	1	1		1 33		1 33
VPP	0	3	3	0	1	1		1 19		1 19
ZOOL	0	41	41	0	3	3		40 46		40 46

IBM 7C94 TABLE III AUGUST, 1968 -CONT

DEPT[1]	NUMBER OF RUNS			NUMBER OF SPECS			IBM 7094 USAGE IN HOURS-MINUTES-SECONDS			
	TANDE[2]	RES	TOTAL	TANDE[2]	RES	TOTAL	TANDE[2]	RESEARCH	REFUND	TOTAL
SUBTOTAL	259	1CC51	10310	12	337	349	14 05 35	365 04 55		379 10 30
DCSSYS[3]	0	464	464	0	8	8		27 11 41		27 11 41
SSUAD[4]	0	132	132	0	2	2		7 32 09		7 32 09
XDCS[5]	0	20	20	0	1	1		14 24		14 24
XSSS[6]	0	64	64	0	1	1		25 51		25 51
SUBTOTAL	0	680	680	0	12	12		35 24 05		35 24 05
TOTALS	259	10731	10990	12	349	361	14 05 35	400 29 00		414 34 30

1 See list of departmental codes following
2 Training and Education
3 System Improvement and Modifications
4 University Administrative Overhead Use
5 System Updating
6 Special Short Shots

Table I - IBM 1401-II

Summary of Use

September, 1968

Scheduled Engineering	2:45
Unscheduled Engineering	18:51
Maintenance	4:17
7094 Preparation	434:17
List/Reproduce	12:15
Tape Dump	2:13
1604 Preparation	1:48
Air Conditioning	5:33
ILLIAC II Preparation	:35
Idle	95:06
Total	577:40

Table II - IBM 1401-II

Summary of Machine Errors

September, 1968

1402 Card Reader Punch	6
Total	6

Table I - IBM 7094

Summary of Use

September, 1968

Scheduled Engineering			32:38
Unscheduled Engineering			4:24
Maintenance			7:21
Air Conditioning			5:49
Idle			144:59
Miscellaneous (Operator training, tape rewind, system tape mounting, rerun of failing problems)			52:26
Total Use			
Training and Education		2:26	
University Administrative Overhead Use		1:39	
System Modification and Improvement		5:51	
System Accounting		7:10	
System Updating		:16	
Customer Use			
In System	235:59		
Relinquish	:37		
Special Short Shots	:40		
Customer Use Total		237:16	
	Total Use		254:38
	Total Time		502:15

Table II - IBM 7094

Summary of Errors

September, 1968

Core Storage		1
Punch		1
	Total	2

IBM 7094 TABLE III SEPTEMBER, 1968

DEPT[1]	NUMBER OF RUNS TANDE[2]	RES	TOTAL	NUMBER OF SPECS TANDE[2]	RES	TOTAL	IBM 7094 USAGE IN HOURS-MINUTES-SECONDS TANDE[2]	RESEARCH	REFUND	TOTAL
AAE	0	63	63	0	2	2		1 51 32		1 51 32
ACCY	0	11	11	0	1	1		6 43		6 43
ADV	0	6	6	0	2	2		26 31		26 31
AGE	0	65	65	0	3	3		32 30		32 30
AGEC	0	144	144	0	6	6		3 05 03		3 05 03
AGREXT	0	5	5	0	1	1		19 12		19 12
AGRON	0	5	5	0	2	2		23 20		23 20
ANS	0	36	36	0	4	4		49 17		49 17
ASTR	0	64	64	0	6	6		1 47 53		1 47 53
ASTRMI	0	1	1	0	1	1		24 57		24 57
BINRES	0	57	57	0	2	2		2 40 04		2 40 04
CE	2	183	185	1	15	16	21	5 10 50		5 11 11
CHE	0	788	788	0	29	29		15 27 04		15 27 04
CIRCE	0	10	10	0	1	1		6 57		6 57
CRC	0	15	15	0	2	2		42 21		42 21
CGS	0	3	3	0	1	1		2 38		2 38
CS	0	2	2	0	2	2		28		28
ECON	0	8	8	0	2	2		15 21		15 21
ED	0	47	47	0	6	6		1 49 28		1 49 28
EDADM	0	3	3	0	1	1		1 58		1 58
EE	0	131	131	0	11	11		3 36 17		3 36 17
ENGCST	0	4	4	0	1	1		6 00		6 00
ENTOM	0	21	21	0	2	2		7 22		7 22
FIN	0	10	10	0	2	2		5 56		5 56
FOR	0	2	2	0	1	1		4 51		4 51
FT	0	8	8	0	1	1		3 25		3 25
GENE	0	9	9	0	1	1		1 26		1 26
GEOL	0	23	23	0	1	1		27 46		27 46
HEC	0	7	7	0	1	1		23 24		23 24
HED	0	10	10	0	1	1		1 12 39		1 12 39
HONORS	0	2	2	0	1	1		1 01		1 01
HORT	0	1	1	0	1	1		43		43
ICR	0	87	87	0	2	2		1 44 52		1 44 52
IED	0	76	76	0	1	1		2 42 32		2 42 32
IGPA	0	5	5	0	1	1		4 04		4 04

IBM 7094 TABLE III SEPTEMBER, 1968 -CONT

DEPT[1]	NUMBER OF RUNS TANDE[2]	RES	TOTAL	NUMBER OF SPECS TANDE[2]	RES	TOTAL	IBM 7094 USAGE IN HOURS-MINUTES-SECONDS TANDE[2]	RESEARCH	REFUND	TOTAL
ILR	0	28	28	0	1	1		5 31		5 31
INADM	0	41	41	0	1	1		27 43		27 43
IREC	0	19	19	0	1	1		1 35 52		1 35 52
LIBS	0	8	8	0	1	1		58 22		58 22
MATH	0	13	13	0	1	1		1 18 57		1 18 57
MATRL	0	715	715	0	21	21		18 41 36		18 41 36
MCBIO	0	29	29	0	3	3		16 29		16 29
ME	14	919	933	1	20	21	19 15	30 50 03		31 09 18
MMPE	0	73	73	0	1	1		1 08 52		1 08 52
NUCE	0	165	165	0	9	9		9 32 08		9 32 08
OIR	0	177	177	0	1	1		6 52 26		6 52 26
PEM	0	48	48	0	7	7		28 38		28 38
PHYB	0	69	69	0	4	4		41 01		41 01
PHYCS	0	208	208	0	5	5		4 45 04		4 45 04
PHYX	0	931	931	0	9	9		86 25 30		86 25 30
PCLS	35	12	47	1	1	2	2 06 28	1 16 55		3 23 23
PSYCH	0	632	632	0	21	21		13 50 55		13 50 55
REC	0	1	1	0	1	1		3 07		3 07
SCENS	0	23	23	0	1	1		9 32		9 32
SGS	0	34	34	0	2	2		13 36		13 36
SCC	0	26	26	0	2	2		1 20 56		1 20 56
SOCW	0	3	3	0	1	1		5 16		5 16
SRL	0	60	60	0	2	2		3 25 47		3 25 47
SWS	0	229	229	0	16	16		3 46 32		3 46 32
TAM	0	37	37	0	4	4		20 22		20 22
VPF	0	2	2	0	1	1		5 20		5 20
VPP	0	2	2	0	1	1		10		10
ZOOL	0	41	41	0	2	2		1 00 57		1 00 57

IBM 7094 TABLE III SEPTEMBER, 1968 -CONT

DEPT[1]	NUMBER OF RUNS			NUMBER OF SPECS			IBM 7094 USAGE IN HOURS-MINUTES-SECONDS			
	TANDE[2]	RES	TOTAL	TANDE[2]	RES	TOTAL	TANDE[2]	RESEARCH	REFUND	TOTAL
SUBTOTAL	51	6457	6508	3	257	260	2 26 04	236 34 02		239 00 06
CCSSYS[3]	C	287	287	0	7	7		13 00 48		13 00 48
SSUAD[4]	C	25	25	0	1	1		1 38 42		1 38 42
XCCS[5]	C	17	17	0	1	1		15 39		15 39
XSSS[6]	0	89	89	0	1	1		40 30		40 30
SUBTOTAL	0	418	418	0	10	10	0 00	15 35 39		15 35 39
TOTAL	51	6875	6926	3	267	270	2 26 04	252 09 41		254 35 45

1 See list of departmental codes following
2 Training and Education
3 System Improvement Modification
4 University Administrative Overhead Use
5 System Updating
6 Special Short Shots

7094 Table III 3rd Quarter, 1968

Dept[1]	Number of Runs			Number of Specs			7094 Usage in Hours-Minutes		
	T and E[2]	Res	Total	T and E[2]	Res	Total	T and E[2]	Res	Total
AAE	0	227	227	0	6	6	0.0	11 2.5	11 2.5
ACCY	0	117	117	0	7	7	0.0	40.6	40.6
ADV	0	112	112	0	12	12	0.0	7 5.8	7 5.8
AGE	0	436	436	0	13	13	0.0	5 42.9	5 42.9
AGEC	0	657	657	0	27	27	0.0	12 21.8	12 21.8
AGREX	0	5	5	0	1	1	0.0	19.1	19.1
AGRON	0	129	129	0	18	18	0.0	4 43.4	4 43.4
ANS	0	187	187	0	13	13	0.0	5 1.7	5 1.7
ANTH	32	26	58	2	2	4	1 58.7	1 5.6	3 4.3
ARCH	6	0	6	2	0	2	4.6	0.0	4.6
ASTR	0	352	352	0	16	16	0.0	8 7.9	8 7.9
ASTRM	0	45	45	0	3	3	0.0	2 55.4	2 55.4
BINRE	0	183	183	0	6	6	0.0	5 20.6	5 20.6
BOT	0	25	25	0	3	3	0.0	14.9	14.9
CCSCS	0	14	14	0	1	1	0.0	11.7	11.7
CE	91	785	876	3	54	57	42.3	16 58.9	17 41.2
CHE	0	2847	2847	0	95	95	0.0	66 35.5	66 35.5
CICBI	0	19	19	0	2	2	0.0	10.2	10.2
CIRCE	0	17	17	0	2	2	0.0	36.2	36.2
CP	0	21	21	0	2	2	0.0	19.4	19.4
CRC	0	37	37	0	7	7	0.0	1 24.5	1 24.5
DCS	0	17	17	0	2	2	0.0	23.8	23.8
DGS	0	3	3	0	1	1	0.0	2.6	2.6
DS	0	31	31	0	6	6	0.0	6.4	6.4
ECON	21	303	324	2	13	15	30.1	2 53.7	3 23.9
ED	0	298	298	0	27	27	0.0	9 53.5	9 53.5
EDADM	0	3	3	0	1	1	0.0	1.9	1.9
EDPSY	11	2	13	2	1	3	20.3	0.8	21.2
EE	0	895	895	0	48	48	0.0	21 47.6	21 47.6
ENGCS	0	5	5	0	2	2	0.0	10.7	10.7
ENTOM	0	65	65	0	6	6	0.0	34.6	34.6
FIN	0	86	86	0	5	5	0.0	1 7.7	1 7.7
FOR	0	31	31	0	5	5	0.0	1 17.3	1 17.3
FT	0	8	8	0	1	1	0.0	3.4	3.4

7094 Table III 3rd Quarter, 1968 Continued

Dept[1]	Number of Runs			Number of Specs			7094 Usage in Hours-Minutes		
	T and E[2]	Res	Total	T and E[2]	Res	Total	T and E[2]	Res	Total
GENE	115	107	222	2	5	7	40.3	1 23.8	2 4.1
GEOG	0	30	30	0	1	1	0.0	22.4	22.4
GEOL	16	197	213	1	7	8	25.5	3 40.2	4 5.8
HEC	0	13	13	0	2	2	0.0	39.0	39.0
HED	0	149	149	0	3	3	0.0	13 45.4	13 45.4
HLTHS	0	21	21	0	1	1	0.0	45.0	45.0
HONOR	0	43	43	0	3	3	0.0	1 21.4	1 21.4
HORT	0	57	57	0	7	7	0.0	52.0	52.0
HOUDI	0	4	4	0	1	1	0.0	10.1	10.1
ICR	0	461	461	0	6	6	0.0	13 1.2	13 1.2
IED	0	242	242	0	3	3	0.0	8 46.6	8 46.6
IGPA	0	57	57	0	3	3	0.0	1 46.1	1 46.1
ILR	0	146	146	0	6	6	0.0	1 46.9	1 46.9
INADM	0	152	152	0	3	3	0.0	2 12.5	2 12.5
IREC	0	117	117	0	6	6	0.0	4 44.0	4 44.0
LIBS	0	13	13	0	3	3	0.0	1 0.4	1 0.4
MATH	0	45	45	0	3	3	0.0	3 56.0	3 56.0
MATRL	0	2827	2827	0	71	71	0.0	93 15.5	93 15.5
MCBIO	0	140	140	0	9	9	0.0	1 4.4	1 4.4
ME	71	2954	3025	6	62	68	33.7	89 58.7	90 32.5
MMPE	0	335	335	0	3	3	0.0	4 56.7	4 56.7
MUSIC	65	147	212	2	2	4	4 35.0	18 10.0	22 45.1
NHS	0	8	8	0	1	1	0.0	23.9	23.9
NUCE	0	963	963	0	28	28	0.0	47 15.1	47 15.1
OIR	0	592	592	0	3	3	0.0	22 11.8	22 11.8
PEM	0	415	415	0	36	36	0.0	3 53.3	3 53.3
PHYB	0	265	265	0	9	9	0.0	4 31.0	4 31.0
PHYCS	0	1038	1038	0	22	22	0.0	19 16.4	19 16.4
PHYSL	0	25	25	0	4	4	0.0	23.8	23.8
PHYX	0	3893	3893	0	29	29	0.0	315 27.7	315 27.7
POLS	193	382	575	5	10	15	12 43.0	20 9.4	32 52.5
PSYCH	0	2967	2967	0	88	88	0.0	64 36.0	64 36.0
REC	0	81	81	0	7	7	0.0	8 56.3	8 56.3
SCONS	0	161	161	0	3	3	0.0	1 5.0	1 5.0
SCS	0	11	11	0	1	1	0.0	5.9	5.9
SGS	0	109	109	0	6	6	0.0	2 46.5	2 46.5

7094 Table III 3rd Quarter, 1968 Continued

Dept[1]	Number of Runs			Number of Specs			7094 Usage in Hours-Minutes		
	T and E[2]	Res	Total	T and E[2]	Res	Total	T and E[2]	Res	Total
SOC	0	310	310	0	16	16	0.0	17 22.6	17 22.6
SOCW	0	29	29	0	3	3	0.0	44.9	44.9
SOLS	0	1	1	0	1	1	0.0	0.5	0.5
SPCH	0	35	35	0	3	3	0.0	43.9	43.9
SPED	0	21	21	0	3	3	0.0	20.7	20.7
SRL	0	275	275	0	6	6	0.0	25 38.7	25 38.7
SWS	0	767	767	0	45	45	0.0	17 23.6	17 23.6
TAM	0	265	265	0	17	17	0.0	2 39.6	2 39.6
VPH	0	6	6	0	3	3	0.0	8.4	8.4
VPP	0	12	12	0	3	3	0.0	3.5	3.5
VTED	0	1	1	0	1	1	0.0	0.2	0.2
ZOOL	0	128	128	0	9	9	0.0	2 8.1	2 8.1
Subtotal	621	29005	29626	27	975	1002	22 34.0	1035 22.9	1057 57.0
DCSSYS[3]	0	1418	1418	0	25	25	0.0	69 49.1	69 49.1
SSUAD[4]	0	245	245	0	4	4	0.0	14 36.4	14 36.4
XDC[5]	0	49	49	0	3	3	0.0	59.2	59.2
XSSS[6]	0	329	329	0	3	3	0.0	2 17.3	2 17.3
Total	621	31046	31667	27	1010	1037	22 34.0	1123 4.9	1145 39.0

1 See list of departmental codes following
2 Training and Education
3 System Improvement and Modifications
4 University Administrative Overhead Use
5 System Updating
6 Special Short Shots

9.2 Research Problem Specifications

During the third quarter of 1968, 51 problem specifications were submitted to the Department for computation on the 7094. The following brief descriptions of these problems have been prepared for inclusion in this report by those submitting them. T indicates a calculation associated with a thesis.

3239-87001 Advertising. Linear Programming in Media Selection. Linear programming model to be used to analyze the different media that can be used in advertising. (John D. Leckenby)

3240-87002 T Electrical Engineering. Beam Steering of Random Arrays with Quantized Phase. Evaluation of the antenna patterns of random phased arrays antennas with quantized phase excitation. (Wells)

3241-87004 Soil Conservation Service. Hydrologic and Economic Analysis of Watersheds. Using the convex coefficient method of flood routing for various frequency storms, and by converting rainfall into runoff, economic analyses are performed. (USDA SCS)

3242-87005 T Civil Engineering. A Systems Analysis of Local Rural Road Administrative Units in Illinois. The thesis is concerned with an operational and economic analysis of the township and district road system in the State of Illinois. The research includes a descriptive statistical analysis of the system and the development and evaluation of proposed operating procedures. (W. D. Berg)

3243-87006 Agricultural Economics. Operating Costs of Country Hog Buying Points. The data for the study has been collected from 47 Illinois country hog markets during the last three years. The analysis will consist of associating cost of firm operation with respect to volume of hogs marketed, and number of men employed. (E. E. Broadbent)

LIST OF DEPARTMENT CODES

If your department or office does not appear on this list, please write its full name in the department field on the Problem Specification Form even though it will require more than 6 characters.

ACCY	Accountancy
ADMREC	Admissions and Records
ADV	Advertising
AAE	Aeronautical and Astronautical Eng.
AGEC	Agricultural Economics
AGE	Agricultural Engineering
AGREXT	Agricultural Extension
AGRON	Agronomy
ANS	Animal Science
ANTH	Anthropology
ARCH	Architecture
ASTR	Astronomy
BIOPH	Biophysics
BOT	Botany
BCMPL	Bureau of Community Planning
BECBSR	Bureau of Economic Business Research
BEDRES	Bureau of Educational Research
BINRES	Bureau of Institutional Research
BED	Business Education
CZR	Center for Zoonoses Research
CERE	Ceramic Engineering
CPUBS	Champaign Public Schools
CHE	Chemistry and Chemical Engineering
CRC	Children's Research Center
CP	City Planning
CE	Civil Engineering
COMM	Communications
CURLAB	Curriculum Laboratory
DS	Dairy Science
DCS	Department of Computer Science
DGS	Division of General Studies
DUE	Division of University Extension
DOW	Division of Waterways
ECON	Economics
ED	Education
EDPSY	Educational Psychology
EDADM	Educational Admin. and Supervision
EDTEST	Educational Testing
EE	Electrical Engineering
ENGADM	Engineering Administration
ENGCST	Engineering College and Station
ENGH	Engineering Honors Program
ENGLSH	English
ENTOM	Entomology
EDC	Extension Division Counseling
FIN	Finance
FT	Food Science
FOR	Forestry
GENE	General Engineering
GEOG	Geography
GEOL	Geology
GER	German
GSBA	Graduate School of Business Admin.
HED	Health Education
HLTHSV	Health Service
HEC	Home Economics
HONORS	Honors Program
HORT	Horticulture
ILLDMH	Illinois Dept. of Mental Health
INADM	Industrial Administration
IE	Industrial Engineering
IREC	Institute for Researh on Exc. Children
ICR	Institute of Communications Research
IGPA	Institute of Govt. and Public Affairs
ILR	Institute of Labor and Ind. Relations
LIBS	Library Science
LING	Linguistics
MKTG	Marketing
MATRL	Materials Research Laboratory
MATH	Mathematics
ME	Mechanical Engineering
MCBIO	Microbiology
MMPE	Mining, Metallurgy, and Petroleum Eng.
MUSIC	Music
NHS	Natural History Survey
NUCE	Nuclear Engineering
OAC	Office of Agricultural Communication
DNW	Office of the Dean of Women
OIR	Office of Instructional Resources
PEM	PE for Men and Graduate PE
PEW	PE for Women
PHYPLA	Physical Plant
PHYCS	Physics
PHYB	Physics Betatron Laboratory
PHYX	Physics Project X
PHYSL	Physiology and Biophysics
PLPA	Plant Pathology
POLS	Political Science
PROVST	Provost's Office
PSYCH	Psychology
REC	Recreation
SHCBRC	Small Homes Council
SOCW	Social Work
SOC	Sociology
SCONS	Soil Conservation Service
SPED	Special Education
PSDEC	Special Education, Decatur Pub. Schools
SPCH	Speech and Theatre
SGS	State Geological Survey
SWS	State Water Survey
SCS	Student Counseling Service
SRL	Survey Research Laboratory
TAM	Theoretical and Applied Mechanics
USGS	U.S. Geological Service
UNIHI	University High School
UCCTE	Urbana-Champaign Coun. on Teacher Ed.
VMS	Veterinary Medical Science
VMA	Veterinary Medicine Administration
VPH	Veterinary Pathology and Hygiene
VPP	Veterinary Physiology and Pharmacology
VTED	Vocational and Technical Education
WPGU	WPGU Radio Station
ZOOL	Zoology

Chicago Circle

CCCHE	Chemistry
CCDME	Materials Engineering
CCENEN	Energy Engineering
CCPHCS	Physics
CCSOC	Sociology
CCSCS	Student Counseling Service

Medical Center

ORME	Office of Research in Medical Ed.
OT	Occupational Therapy

Ill. State University

ISEDAD	Department of Education Administration

3244-87007 Vocational and Technical Education. Achievement Measures Project. Item analyses and factor analyses of experiment forms of achievement test in eight vocational areas. (Baldwin)

3245-87008 T Physical Education for Men and Graduate PE. The Influence of Laterality on Girls' Ability to Throw a Softball. A one-way analysis of variance in attempt to determine whether one particular laterality pattern has a significant influence on girls', age 13 and 14, ability to throw a softball for distance. (Barbara Todd)

3246-87011 T Physical Education for Men and Graduate PE. Achievement Orientation and Its Influence on the System of the School Class. Six experimental groups and five control groups were measured by means of a sociometric questionnaire to determine the effects, if any, on group structure of an experimental variable of high achievement orientation. One hundred and sixty-five matrices were generated from the raw data. From this will be obtained indices of reciprocal relationships, rank correlations, two way ANOVA within groups and between groups on the various questions, and on the grouped experimental and control units. (G. M. McKelvey)

3247-87012 Natural History Survey. Effects of Chelating Agents and Metal Chelates on Insects. The toxicological and physiological effects of chelating agents and chelated metals, expecially copper (II) are being studied. Such parameters as mortality, fecundity, fertility, and metabolic abnormalities are utilized. (Douglas K. Sell)

3248-87013 T Psychology. Multiple Discriminant Analysis of Perceptual Form Constancy Responses. A multiple discriminant analysis of each of about 110 subject's responses to 25 presentations of a set of nine stimuli in a form-constancy experiment. The output of these analyses will then be subjected to analyses of variance, correlations, and other statistical tests. (Dale Kaess)

3249-87014 Children's Research Center. Hyperactivity in Children. Extensive use of SSUPAC. (R. L. Sprague)

3250-87016 School of Life Sciences. Speciation in Western Mammals. The species-status of two populations of pocket gophers is being tested. Problems in speciation and numerical taxonomy are involved. (Hoffmeister)

3251-87018 General Engineering. Pictorial Drawings. Generation of limited pictorial drawings for ultimate use in graphics courses. (M. H. Pleck)

3252-87019 T Accountancy. Leadership Style and Personality Characteristics in Performance Appraisal. Statistical analysis of data related to raters' personality, leadership style, and rating pattern in a program of performance appraisal. (Abou Elenein Gaber)

3253-87020 Geology. Oceanographic Data Reduction. Marine geology data will be reduced and graphically plotted. (A. F. Richards)

3254-87021 Psychology. Verbal Learning. The time will be used to analyze data from verbal learning research. Analysis of variance will be the primary statistic used. (Montague)

3255-87022 Entomology. Statistical Analysis for Entomology. Routine statistical analysis of experiments in entomology. (R. B. Selander)

3256-87023 T Sociology. Hennepin Project - Mental Health Data. SSUPAC programming for the analysis of data relating to the mental health of survey research respondents in five Illinois counties. Special emphasis to be given to measures of significance, association, and factor analytic techniques in analyzing data similar to Srole's Midtown Manhattan Study. (Lauren Seiler)

3257-87024 T Education. Study of Role Perception. Responses of superintendents, principals, and curriculum directors to critical incidents relating to curriculum, instruction, staffing, program evaluation, etc., give perceptions of the role of curriculum directors in public school systems. Responses are in terms of six variables per incident using a scale with values 0 - 4. (Bryon Shinn)

3258-87025 Recreation. Social and Psychological Correlates of Leisure. Multivariate analyses of social and psychological characteristics associated with various forms of leisure activity. Data is obtained primarily from large-scale surveys of different communities. Analyses will be done using the SSUPAC system primarily but also a few specially written FORTRAN programs. (Doyle Bishop)

3259-87026 T Political Science. Public Opinion and Vietnam Policy. The purpose of the study is to determine the correlates of attitudes toward United States policy in Vietnam in the 1950's. Using a set of ten public opinion surveys conducted by the Roper Center, attitudes toward United States involvement in Vietnam will be correlated with various standard socio-economic variables and with attitudes toward Communism, voting behavior and other relevant variables. SSUPAC programs will be used, in particular the measures of association and analysis of variance. (Susan Welch)

3260-87027 T Communications. Japanese-American Acculturation Research. The problem is an empirical study of the influences of communicational interactions upon acculturation of an ethnic group in U. S. society. (Kiyoshi Nagata)

3261-87028 Psychology. Reactions to Risk and Stress. An exploration of the effects of social comparison processes and personality and attitudinal variables on risk-taking behaviors. (J. McGrath)

3262-87029 T Psychology. The Effects of Penalty and Attentional Manipulations on Children's Discrimination Learning. This research looks at the effects of penalty, instructions, and age on the ability of young children to solve problems involving figure discrimination. Children of two different ages were exposed to different conditions of reward and penalty. Some children were penalized for incorrect responses by the removal of pennies that they had won and other children were not penalized. In addition, some children received special instructions to help them solve the problems. (Susan Taub)

3263-87030 Children's Research Center. Hyperactivity in Children. Extensive use of SSUPAC to determine the levels of physical activity on the part of children. (R. L. Sprague)

3264-87031 Education. Correlation Study - National Teacher Exams and G.P.A. The study will yield simple and multiple correlations and partial and composite scores on the National Teachers Examination with cumulative, transfer, University of Illinois, professional, and major field grade point averages. (R. E. Williams)

3265-87032 Finance. Growth Rates of Nonlife Insurance Companies. Calculation of 840 separate r's (average compound percentage rate of growth) of admitted assets for a random sample of 210 nonlife insurers. The A's and P's have been calculated. The following formula will be used, with 420 calculations involving a nine-year period (n=9) and 420 calculations involving a 12-year period (n=12)--

$$r = \left[\sqrt[n]{\frac{A}{P}} \right] - 1$$

where A = ending admitted asset figure for insurer and P = beginning admitted asset figure for insurer. The A's, P's, and n's will be supplied; the computer is to solve 840 of these equations for r. (Stephen W. Forbes)

3266-87033 T Recreation. Movement Patterns of Children in a Play Area. This problem involves the study of movement patterns of 3- and 4-year old children in a play area. Changes in movement patterns are to be studied over succeeding play sessions relative to fixed play apparatus. The children's positions over 10-second intervals have been plotted by means of Cartesian coordinates, thus permitting an identification of the children's changes in position relative to one another and to the apparatus. The children's paths of movement, their frequency in various areas of the play area, and mean distances between children will be computed and graphed over time. (Wuellner)

3267-87034 Sociology. Control of Youth in Japan. Standard SSUPAC programs will be used to analyze relationships among nine social control agencies in Tokyo. N = 668. Basic descriptive data for all agencies will first be run and then relationships between selected independent variables (agency, level of personnel in agency, image scores, etc.) will be run against dependent variables (i.e., quality and quantity of interaction, operating philosophies). (John Clark)

3268-87035 Student Counseling Service. Programmed vs. Face to Face Psychotherapy. A comparison of effectiveness of using written programmed materials to assist students to overcome test anxiety and of using counseling interviews. Certain test devices are being studied for use in evaluation of the therapies. These include a test anxiety test, a test of interest and motivation, and a test of personal self-evaluation. Statistical procedures will include correlation, factor analysis, and comparisons of means of different groups. (Thomas N. Ewing)

3269-87036 T Sociology. Religiosity and Deviant Attitudes. Analysis of the relationship between the importance of religion to persons, their religious participation and their tendency to hold attitudes deviating from those of their community, in terms of alienation, political affiliation, attitudes toward minority groups, etc. (Richard L. Hough)

3270-87037 T Sociology. Social Engagement in Political Affairs According to Family Life Cycle Status. I propose to determine whether adult family members in different phases of a "family life cycle" differ in their engagement in national political events. Engagement will be measured in terms of the adult family member's interest in, knowledge of, and participation in political events. A secondary analysis will be made of data collected for two national surveys by the Survey Research Center. These studies are the 1966 and the 1956 Election Studies. (Donald Conover)

3271-87038 T Accountancy. An Empirical Test to the Motivation-Hygiene Theory. An empirical test to Herzberg's motivation-hygiene theory of job satisfaction. One group of subjects will be tested using two different methodologies, one of them is Herzberg's. The similarities between the results would support Herzberg's theory while dissimilarities would not. (Soliman Hanafi)

3272-87039 T Psychology. Levels of Word Association and Reversal and Nonreversal Shifts. SSUPAC will be used for several analyses of variance and some correlations. (D. McGrath)

3273-88001 Center for Instructional Research and Curriculum Evaluation. EPDA Proposal Review and Evaluation. Evaluation of questionnaires administered to reviewers of proposals submitted to U. S. government educational funding agency. (Thomas J. Hastings)

3274-88002 T Special Education. Assessing Congruence of Intent and Practice in Instruction. Instructional objectives and test questions are classified on a Taxonomy of Cognitive Abilities by instructors and other raters. Data on rater reliability and variability are obtained through two-way frequency counts, contingency tables, and rater rank order correlations. Level of emphasis on cognitive abilities defined by the Taxonomy is assigned for each instructor based on a weighted key. The level of emphasis for objectives and questions are compared and serve as indices of congruence of intent and practice. (Joe M. Steele)

3275-88003 T Educational Administration. Learning Resources Center Survey. This is an evaluation of the effectiveness of the Learning Resources Center concept in the elementary school (K-5). Data will be used to compare means of the four resource groups (teachers, parents, students, and principals) by t test (pooled variance). A comparison of means of concepts will be made. (Gundy Gerdon)

3276-88004 Physical Education for Men and Graduate PE. Serial Analysis of Variance for Repeated Measures. Successive trial-by-trial (serial) analyses of variance are proposed as a possible method for deriving a terminal variance estimate as a basis for comparing inherent treatment variance and as a basis for post hoc comparisons to determine when improvement ceased and random variation remained. (A. W. Hubbard)

3277-88005 Veterinary Pathology and Hygiene. Parasitic Diseases of Cattle. Obtain information concerning the effects of the wanderings of abberrant parasites on the general health of cattle. Normal parasites of cattle are also studied. (Fitzgerald)

3278-88006 T Education. Demographic Characteristics of Institutionalized Disturbed Children. Study characteristics and relationships among characteristics of emotionally disturbed children in mental hospitals. (Hasbargen)

3279-88007 T Economics. Sales Tax Shifting in Selected SMSA's. The problem is to measure the effect of state sales tax rate changes on selected metropolitan consumer price indexes. These indexes are computed inclusive of sales taxes. The object is to discover whether and to what extent retailers actually do shift the burden of sales taxes onto their customers. (Nancy Sidhu)

3280-88008 T Education. Study of Attitudes and Approaches to Writing. The study seeks to ascertain whether there are significant differences between selected attitudes and approaches to composition between writers of superior performance and average performance. (LaCampagne)

3281-89004 Economics. Statistical Analysis of German Public Debt Structure. Test of the relations between certain interest rates and the structure of public debt in Germany. (Willms)

3282-89008 Education. Evaluation Undergraduate Education Courses. This research is aimed at evaluating the extent to which objectives of professional courses in the undergraduate teacher education sequence are being achieved. Data were obtained from students enrolled in courses and from graduates of the program. Frequency counts, means, standard deviations, and differences of means tests will be compiled. (Harold H. Lerch)

3283-89011 Health Service. Incidence of Psychiatric Disorders in Students. Tabulation of variables from Health Service patient punch cards. (Orville S. Walters)

3284-89017 T Psychology. Job Meaningfulness and Relationships with Job Satisfaction. The purposes of this study are as follows: (1) the systematic administration of Osgood, et al's semantic differential to certain facets of a job situation; (2) the application of a measure of satiability in the same job situation; and (3) the investigation of relationships between the factors of the semantic differential and certain measures of job satisfaction. (Lewin)

3285-89018 Psychology. Repetition in Behavior. This project deals with an examination of the effects of repetition in behavior under a number of conditions. Independent conditions include auditory presentation of recorded materials and visual presentation of filmed events. Dependent measures include rating scales and evaluation questionnaires. (Jakobovits)

3286-89023 Provost's Office. Follow Study of Transfer Students to University of Illinois. Group and sub-group summaries and comparisons and statistical analyses such as correlations and step-wise multiple regression. (Ernest F. Anderson)

3287-89025 Office of Research in Medical Education. Study of the effects of training conditions on performance in Orthopaedic residency training. (Thomas Bligh)

3288-89026 Psychology. Preference for Novelty in Infants. Infants above 8 weeks of age show more interest in novel than in familiar objects. We are interested in whether or not they go through a prior stage of attachment to familiar objects. Such a finding would have many implications for theories of cognitive development. We have tested the relative preferences for novel or familiar objects (using visual fixation times as our measure) of a group of infants at six and again at eight weeks of age. We have predicted a shift from preference for the familiar to preference for the novel with increasing age. A second group of infants will be tested only once (at eight weeks) to control for the influence of the testing procedure on the six week babies. (Fredric Weizmann)

3289-89029 Institute for Research on Exceptional Children. Use of Color with Retardates. Color vision status with mentally retarded children. (Salvia)

On July 12, 1968, the Department announced that the 7094 computer would be withdrawn from service June 30, 1969. After July 15, 1968, no 7094 problem specifications numbers were issued for new projects, except for those projects which required a facility which was available on the 7094 but not available on the System/360. The most important such facility is SSUPAC.

9.3 Class Problem Specifications

During the third quarter of 1968, 17 problem specifications were submitted to cover all assigned problems on the 7094 in the following courses.

J938-87003	Economics 470.
J939-87009	Music 320.
J940-87010	General Engineering 010.
J941-87015	Civil Engineering 322.
J942-87017	Architecture 494.
J943-89002	Architecture 347.
J944-89009	Psychology 409.
J945-89010	Psychology 494.
J946-89012	Educational Psychology 449.
J947-89016	Civil Engineering 361.
J948-89019	Agricultural Economics 341.
J949-89020	Agricultural Economics 325.
J950-89021	Economics 476.
J951-89022	Agricultural Economics 325.
J952-89024	Social Work 491.
J953-89027	Political Science 430.

10. GENERAL DEPARTMENT INFORMATION

10.1 Personnel

The number of people associated with the Department in various capacities is given in the following table:

	Full-time	Part-time	Full-time Equivalent
Faculty	23	3	24.30
Visiting Faculty	1	0	1.00
Research Associates	4	0	4.00
Graduate Research Assistants	13	105	68.04
Graduate Teaching Assistants	0	8	3.00
Professional Personnel	26	3	28.17
Administrative and Clerical	25	0	25.00
Nonacademic Personnel (Monthly)	64	3	65.75
Nonacademic Personnel (Hourly)	114	1	36.46
TOTAL	270	122	255.72

The Department Advisory Committee consists of Professor J. R. Pasta, Head of the Department; Professor J. N. Snyder, Associate Head of the Department; Professors K. W. Dickman, M. Faiman, L. D. Fosdick, H. G. Friedman, C. W. Gear, D. B. Gillies, D. J. Kuck, B. H. McCormick, S. Muroga, T. A. Murrell, J. Nievergelt, R. S. Northcote, J. R. Phillips, W. J. Poppelbaum, S. R. Ray, J. E. Robertson, P. E. Saylor, and D. L. Slotnick.

10.2 Bibliography

During the third quarter, the following publications were issued by the Laboratory:

File Numbers

(1) Ackins, G. M., "Successive Vector-Element Processes and Variations on the Sorting Problem," File No. 765, July 11, 1968.

(2) Ackins, G. M., "Fast Fourier Transform," File No. 768, July 12, 1968.

(3) Borovec, Richard T., "The Pattern Articulation Unit of Illiac III: Display System: Design and Implementation," File No. 776, September 30, 1968.

(4) Katz, Michael L., "Mathematical Subroutines forIlliac IV," File No. 766, July 11, 1968.

(5) Kerkering, T. E., "Results of Tests Performed on Two Different Types of False Floor Panels Used in DCL 194 and DCL 127," File No. 774, July 29, 1968.

(6) Krabbe, S. Paul, "Specifications for Termi-Point Wiring of Sections 2 and 4 (Transfer Memory) of the Illiac III Computer System," File No. 550-114, July 2, 1968.

(7) Krabbe, S. Paul, "Specifications for a 9" False Floor for DCL 223 That Overlaps the Existing False Floor in DCL 280," File No. 773, August 7, 1968.

(8) Krabbe, S. Paul, "Supplemental Specifications for Wire-Wrap Wiring of Sections 1 and 3 (PAU) of the Illiac III Computer System," File No. 550-116, August 13, 1968.

(9) McCarthy, T., "Solution of Ordinary Differential Equations Related to Metabolic Systems," File No. 767, July 11, 1968.

(10) Rollenhagen, David, "Specifications for CRT-Type Display System with x and y Inputs and Variable Persistence," File No. 550-115, July 25, 1968.

(11) Schwebel, John, "Use of Graph Transformations to Character Characterize an Image: An Illustrative Example," File No. 769, July 18, 1968.

(12) Schwebel, John and McCormick, B. H., "Properties of a Discrete Space Preserved by Image Processing Relations," File No. 770, July 18, 1968.

(13) Serio, F. P. and Pelg, E., "Procurement and Design Specifications for Printed Wiring Boards," File No. 550-117, September 23, 1968.

(14) Westlund, G. A., "A Timing Simulator," File No. 775, September 6, 1968.

(15) Winje, Gilbert L., "Implementation of the "Monte Carle Evaluation of the Boltzmann Collision," File No. 772, August 1, 1968.

Theses

(1) Whitaker, Willard Francis, III, "Design Procedures for Cellular Logic Arrays," (M.S.), Report No. 288, September 20, 1968.

10.3 Colloquia

"Numberical Solution of the Navier-Stokes Equation," by Mr. James Pestaner, U. S. Naval Radiological Defense Laboratory, San Francisco, California, August 15, 1968.

"Recent Work in Three-Dimensional Displays," by Professor Ivan E. Sutherland, Aitken Computer Laboratory, Harvard University, Cambridge, Massachusetts, August 19, 1968.

"Recent Work in Meteorology," by Professor Yoshimitsu Ogura, University of Tokyo, Nakano, Tokyo, Japan, September 4, 1968.

10.4 Drafting

During the third quarter, a total of 510 drawings were processed by both drafting sections:

	General	Pattern Recognition
Large Drawings	24	64
Medium Drawings	117	49
Small Drawings	23	116
Layouts	1	0
Report Drawings	44	0
Changes	45	26
Miscellaneous	1	0
TOTAL	255	255

(M. Goebel and J. Otten)

10.5 Shops' Production

Job orders processed and completed during the third quarter of 1968 are as follows:

	AEC 1018	AEC 1469	Other
Machine Shop	15	16	14
Electronics Shop	41	42	0
Chemical Shop	15	45	14
Layout Shop	36	41	2
TOTAL	107	144	30

Wiring of 369 printed circuit boards during this period accounted for 12,622 transistors, 18,300 diodes, and 1,375 integrated circuits.

(F. P. Serio)

Physics

COO-1469-0107
COO-1018-1170

QUARTERLY TECHNICAL PROGRESS REPORT

October, November, December 1968

DEPARTMENT OF COMPUTER SCIENCE
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN · URBANA, ILLINOIS

QUARTERLY TECHNICAL PROGRESS REPORT

October, November, December 1968

Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, Illinois 61801

TABLE OF CONTENTS

TABLE OF CONTENTS (CONT'D)

TABLE OF CONTENTS (CONT'D)

TABLE OF CONTENTS (CONT'D)

TABLE OF CONTENTS (CONT'D)

TABLE OF CONTENTS (CONT'D)

TABLE OF CONTENTS (CONT'D)

1. CIRCUIT RESEARCH PROGRAM

(Supported in part by the Office of Naval Research under Contract NR 048-102/6-15-67 Code437, W. J. Poppelbaum, Principal Investigator).

Summary

John Esch has designed a general purpose computing "element" for the Stochastic Computing project. Under digital control it can generate any of the functions: a, b, a^2, b^2, a + b, a - b, a x b, a/b, and is based on design equations described in an earlier report.

Potentiomatrix has been completed and its perpetrator, Bill Steiner, has departed, certificate in hand, for the fleshpots of Lockheed, Georgia. His summary is based on the more detailed, forthcoming D.C.S. Report No. 307.

Finally, David Ring discusses a number representation that might be useful to the work on Bundle Processing.

M. Faiman

1.1 Stochastic Computing (Project No. 03)

1.1.1 Design of a General Computing Element

In the Quarterly Technical Progress Report for January - March 1968, section 1.2.4, page 8, equations were derived for performing addition, subtraction, multiplication and division on CRPS's, using the two-wire ratio representation. For convenience, we summarize the results of that derivation.

A rational number, a ($|a| < 1$), is encoded into two CRPS's, A_n and A_d, representing numerator and denominator, respectively. Z_1 and Z_2 are two independent (statistically uncorrelated) representations of zero. Table 1 describes the required operations.

	Numerator	Denominator
Product	$P_n = A_n \oplus B_n$	$P_d = A_d \oplus B_d$
Quotient	$Q_n = A_n \oplus B_d$	$Q_d = A_d \oplus B_n$
Sum	$S_n = Q_n Z_1 \vee Q_d \overline{Z}_1$	$S_d = P_d Z_1 \vee Z_2 \overline{Z}_1$
Difference	$D_n = Q_n Z_1 \vee \overline{Q}_d \overline{Z}_1$	$D_d = S_d$

Table 1. Formulas for Arithmetic Operations

We require a stochastic computing element, having two CRPS inputs representing the numbers a and b, say, and three digital, controlling signals F, O and I, which generates a result, r, according to Table 2.

F O I	r
0 0 0	a
0 0 1	b
0 1 0	a^2
0 1 1	b^2
1 0 0	a + b
1 0 1	a - b
1 1 0	a x b
1 1 1	a / b

Table 2. Coding for Stochastic Computing Element.

If the result is encoded into the two CRPS's R_ϵ, where ϵ = n, d, then evidently.

$$R_\epsilon = \overline{F}\overline{O}U_\epsilon \vee \overline{F}OU^2_\epsilon \vee \overline{F}O +_\epsilon \vee \overline{F}O\, x_\epsilon \tag{1}$$

and the intermediate Boolean functions U_ϵ, U^2_ϵ, $+_\epsilon$ and x_ϵ are given by

$$U_\epsilon = \overline{I}A_\epsilon \vee IB_\epsilon \tag{2}$$

$$U^2_\epsilon = U_\epsilon \oplus \Delta U_\epsilon \tag{3}$$

$$+_n = \overline{Z}_1(I \oplus Q_d) \vee Z_1Q_n \tag{4}$$

$$+_d = \overline{Z}_1Z_2 \vee Z_1P_d \tag{5}$$

$$x_\epsilon = \overline{I}P_\epsilon \vee IQ_\epsilon \tag{6}$$

In equation (3), ΔU_ϵ denotes a delayed version of U_ϵ. A network to realize R_n and R_d is shown in Figure 1.

John Esch

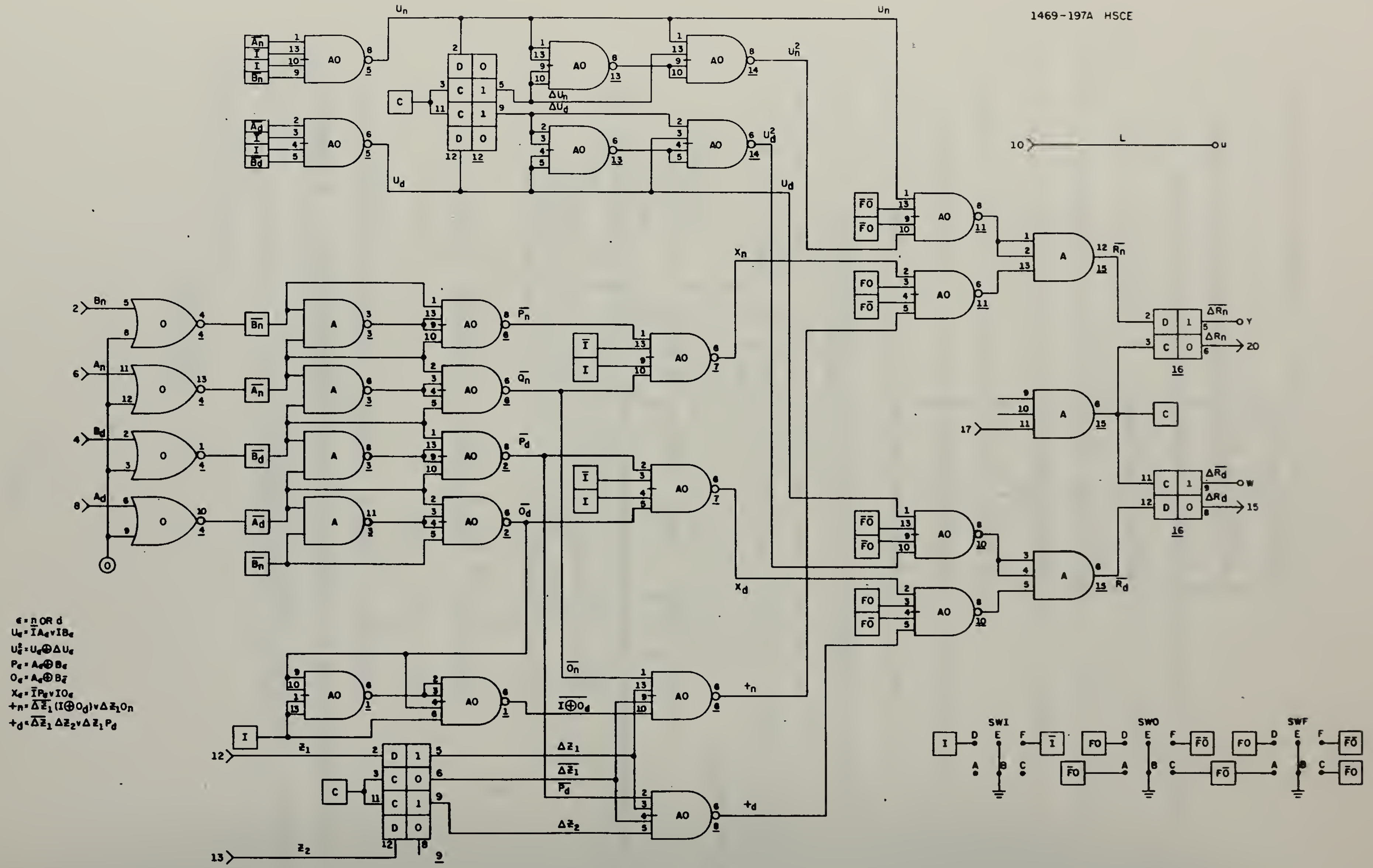


Figure 1. Logic of General Stochastic Computing Element.

1.2 Potentiomatrix (Project No. 08)

1.2.1 Introduction

Potentiomatrix is an electronic system utilizing hybrid digital-analog circuitry for displaying an arbitrary family of equipotential lines. It uses the principle that certain areas clamped to fixed voltages in an infinite conducting plane give rise to a family of equipotential lines which are governed by the two dimensional Laplace's differential equation with appropriate boundary conditions.

It is somewhat helpful to think of the Potentiomatrix panel as representing a section of a grid of resistors which imitates the behavior of an infinite conducting plane. The panel lamps then represent nodes in the grid where four resistors are connected. Each of the 32 x 32 (1024) nodes in the central portion of the grid is connected to a pair of electronic circuits which are in turn connected respectively to a lamp in the panel and to a metal lamp socket which appears on the panel as a conducting ring around the lamp.

The circuits which are connected to the conducting rings through their inputs are the write circuits and they are used to clamp the appropriate nodes to the proper voltage values ($+10 = V_{B1}$, $-10v = V_{B2}$). The use of the write circuits corresponds to applying boundary conditions to Laplace's differential equation. Touching the brush pen to the conducting ring applies $\pm$10v (through 510 ohms) to the input of the write circuit which applies V_{B1} or V_{B2} to the node in the grid. Touching 0v, which corresponds to a selective erase, resets the circuit and allows the node to float. In addition to these individual controls, there are two common busses, R+ and R-, which perform master erase functions by

resetting all of the write circuits.

The circuits whose outputs are connected to the lamps are the <u>detector</u> circuits and they are used to compare the voltages at their nodes in the grid with a common reference signal, V_S, which goes to all the detector circuits. If a node voltage is equal to the reference, V_S, within a preset tolerance E, the detector circuit, in conjunction with a clocking signal G, gates on the lamp for that node. It is evident that the lamps correspond to nodes in the grid which are each approximately at the value of the reference voltage, V_S, and therefore display an equipotential line.

The reference voltage, V_S, takes on successively the analog values V_{B1}, V_{B2}, V_{F1}, V_{F2}, V_{F3}, V_{F4} and V_{F5}. This is pictured in Figure 1. V_{B1} and V_{B2} are the voltage levels ($\pm$ 10v) which generate the equipotential lines corresponding to the boundary conditions, and V_{F1} through V_{F4} are arbitrary voltage levels between V_{B1} and V_{B2} which are selected by the operator and which correspond to four intermediate members of the equipotential family. V_{F5} has a slightly different effect, which will be discussed in the next section. As the reference voltage, V_S, proceeds through its seven values, the lamps remain on so that each new equipotential line is displayed along with the old ones. After the seven lines have been displayed for a short time, a buss signal R resets all of the detector circuits, thereby turning the lamps off, and the cycle repeats.

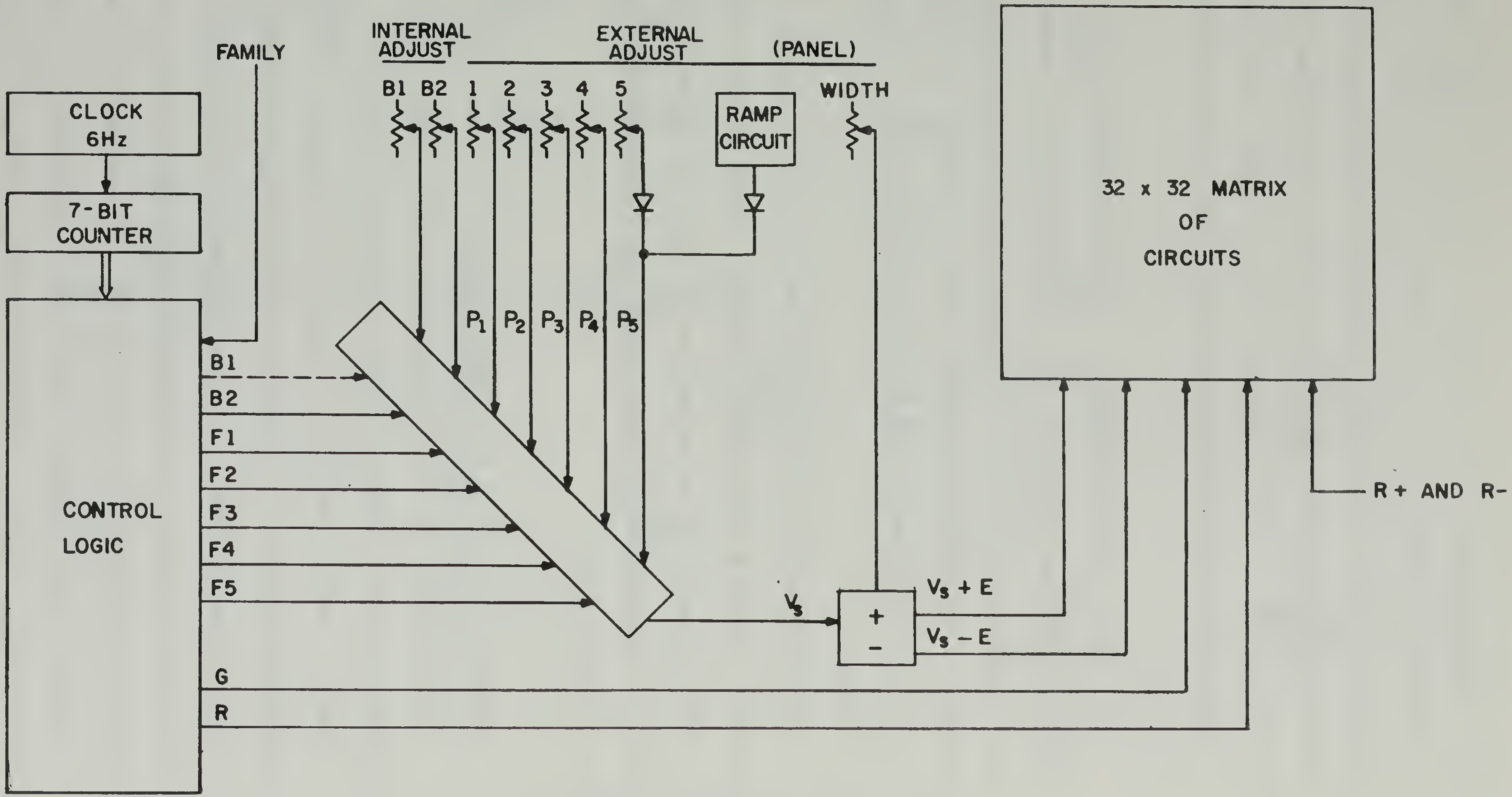


Figure 1. Functional Diagram of Potentiomatrix

1.2.2 Ramp Circuit

In order to add motion to the Potentiomatrix display, a ramp generator has been added to the control circuitry in conjunction with V_{F5}. This circuitry allows the operator to place a time varying voltage as one analog value on the reference voltage, V_S, which has the effect of dynamically sweeping through any desired set of equipotentials on the display.

The circuit, shown in Figure 2, consists essentially of a unijunction relaxation oscillator and two amplifiers for increasing the signal amplitude and drive. The three potentiometers, 500K, 50K, and 5K adjust the frequency, amplitude, and d.c. level of the signal respectively. The period of the signal is typically between .5 seconds and 2.5 seconds.

The output of this circuit is joined in an "or" circuit with the wiper on the F5 control potentiometer. Since an "or" circuit for analog signals selects the maximum of its input signals, the output of the ramp circuit can be controlled by the operator with the F5 potentiometer.

1.2.3 Boundary Conditions for Common Shapes

The usefulness of the Potentiomatrix system depends on the ease with which the shape of curved lines may be transmitted. Although no attempt is made to catalog a complete set of curves, a few of the more obvious ones must be listed so that the simplicity behind this technique becomes evident. In order to demonstrate the simple conditions needed to specify a desired curve, the complete set of conic sections is considered in these examples. The available voltages to which points in the conductive sheet may be fixed are V_{B1}, a positive boundary voltage, and V_{B2}, a negative boundary voltage of the same

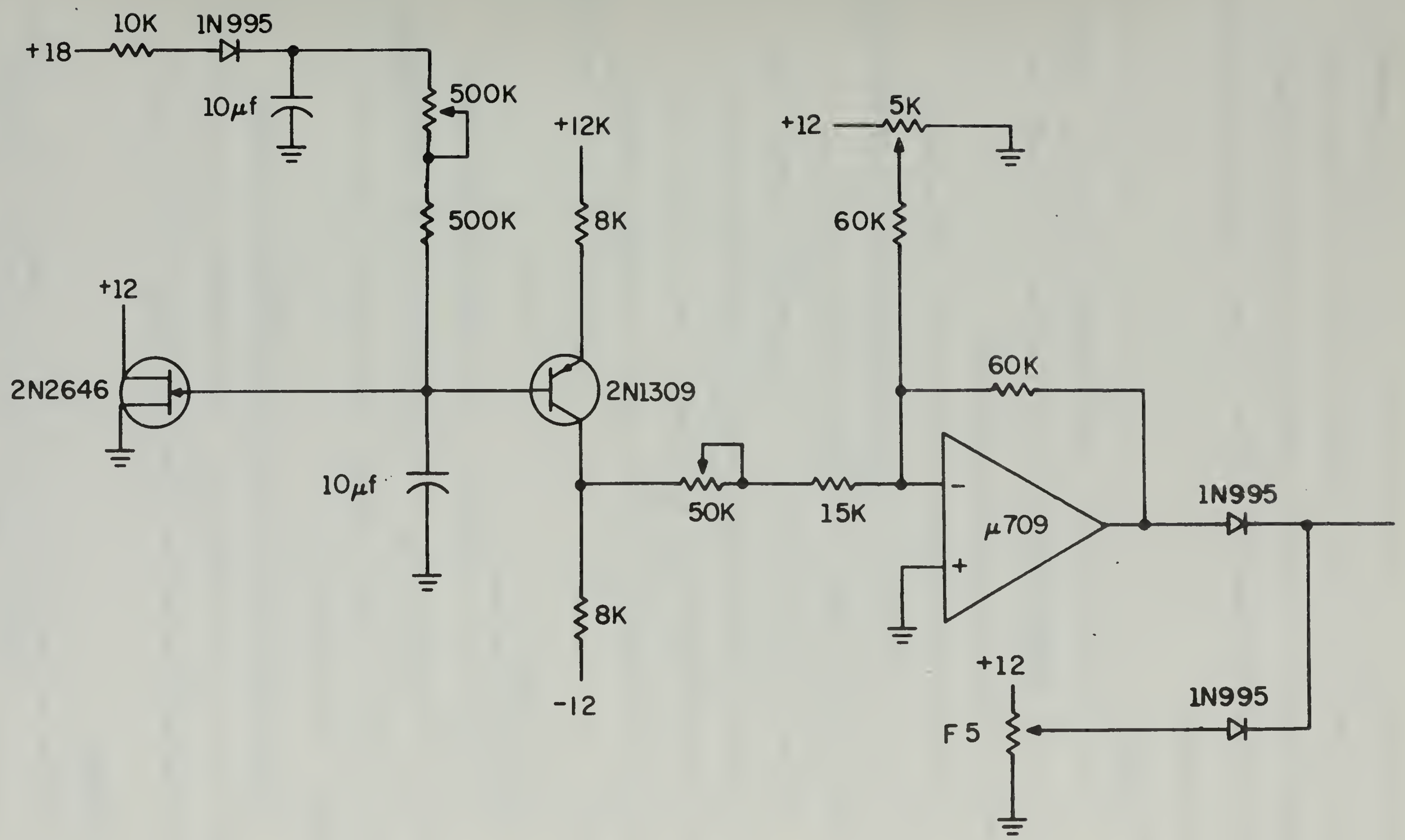


Figure 2. Ramp Circuit

magnitude as V_{B1}. The curves generated by these arbitrary boundary voltages and the fixed boundary voltage at infinity are formed as equipotential lines in a conductive plane and are governed by Laplace's two-dimensional differential equation. Figure 3 accompanies the following examples.

(a) Circle

Circles are generated by a single boundary point at V_{B1}. Using this input point as a center, concentric circles of all radii are formed for voltages between V_{B1} and ground.

Circles are also generated by two separate boundary points, one at V_{B1}, and the other at V_{B2}. In this case, the circles are not concentric and are formed for voltages between V_{B1} and V_{B2}.

Circles are generated in still a third way by a straight line at V_{B1} and a non-colinear point at V_{B2}. In this case, the point forms an electrostatic image on the other side of the line. The circles are not concentric but are formed for voltages between V_{B1} and V_{B2}.

(b) Straight Line

A straight line is generated by two separate boundary points, one at V_{B1}, the other at V_{B2}. The straight line is the perpendicular bisector of a line segment joining the two boundary points and is formed at ground potential.

Straight lines are also generated by two parallel straight lines, one at V_{B1}, the other at V_{B2}. The straight lines are parallel to the boundary lines and are formed for voltages between V_{B1} and V_{B2}.

(c) Ellipse

Ellipses are generated by a straight line segment at V_{B1}. Using the end points of the line segment as foci, ellipses are formed for

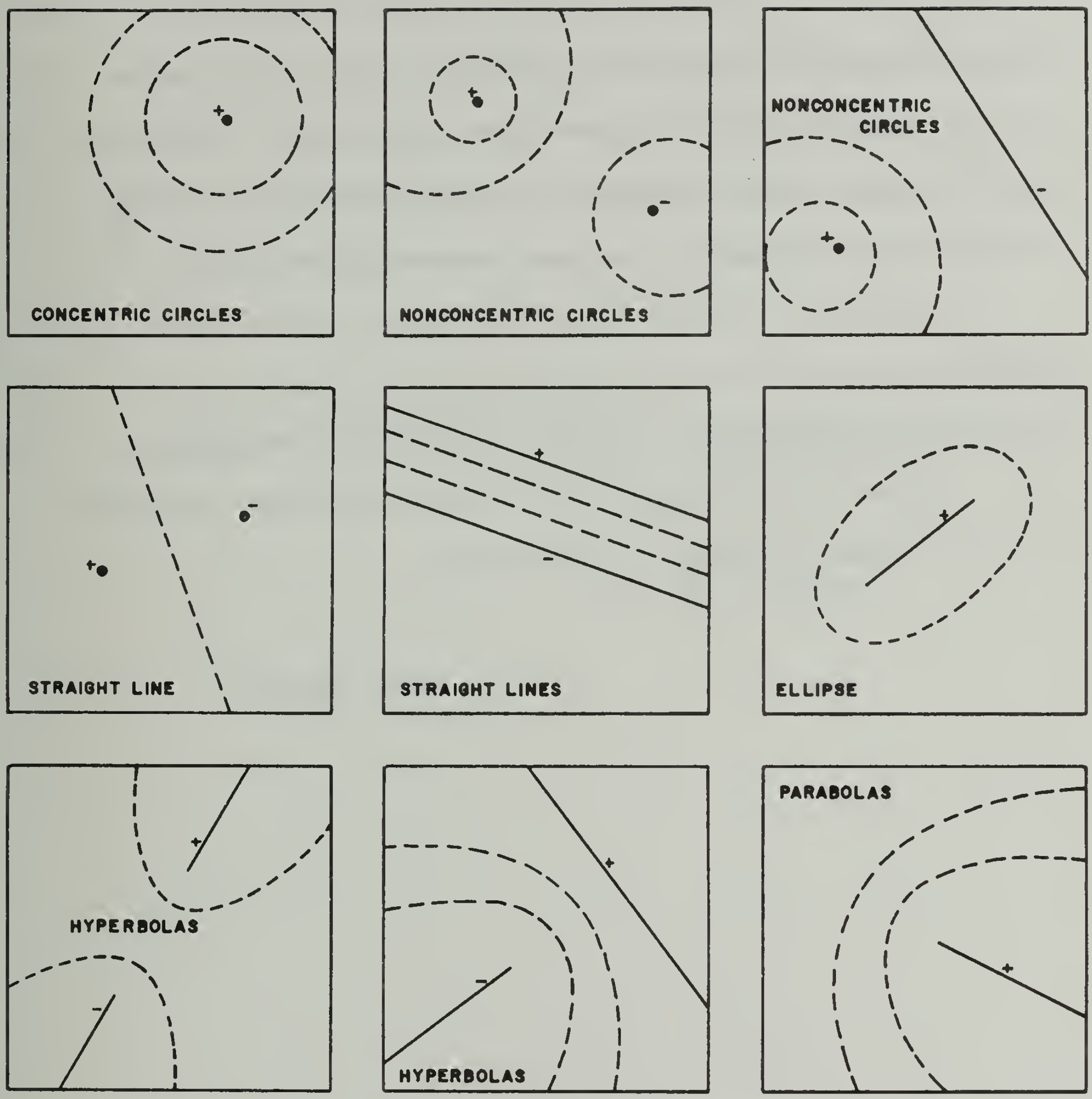


Figure 3. Boundary Conditions for Conic Sections

voltages between V_{B1} and ground.

(d) Hyperbola

Hyperbolas are generated by a pair of semi-infinite colinear lines, one at V_{B1}, the other at V_{B2}. Using the end points of the lines as foci, hyperbolas are formed for voltages between V_{B1} and V_{B2}.

Hyperbolas are also generated by a straight line at V_{B1} and a semi-infinite non-intersecting perpendicular line at V_{B2}. In this case an electrostatic image is formed on the other side of the V_{B1} line. Hyperbolas are formed for voltages between V_{B1} and V_{B2}.

(e) Parabola

Parabolas are generated by a single semi-infinite straight line at V_{B1}. Using the end point of the line as the focus, parabolas are formed for voltages between V_{B1} and ground.

Bill Steiner

1.3 Large Screen TV Display (Project No. 09)

1.3.1 The Problem of Modulation and Deflection

Investigations are being made to determine the most suitable techniques for modulating and deflecting a light beam to be used for a large screen display. Three types of modulator have been studied so far, based respectively on interference, the Kerr/Pockels electro-optic effects and acousto-optic modulation. The last appears to be the most promising in terms of power consumption, reliability and cost.

The deflection problem is more critical since it involves questions of synchronization, resolution and linearity. A mechanical solution for the relatively slow vertical scan rate may be attractive, but a mechanism for the 15.75 kHz horizontal sweep has yet to be found.

M. N. Cooper

1.4 Bundle Processing (Project No. 21)

1.4.1 Number Representation

A two - bundle representation of rational numbers is being investigated, described by equations which are essentially similar to those of Table 1 in section 1.1.1 of this report. Not yet solved, however, is the "addition - attrition" problem in which successive additions cause the average number of active wires to be drastically reduced.

David Ring

2. HARDWARE SYSTEMS RESEARCH

(Supported in part by the Atomic Energy Commission under Contract US AEC AT (11-1) 1949, W. J. Poppelbaum, Principal Investigator.)

Summary

David Casasent, Doug Sand and Al Irwin describe resolution tests and equipment modifications on OLFT. Full details of the variable scan rate feature of VISTA and of the vidicon circuitry are given by David Rollenhagen. In Functional Encoding, Peter Oberbeck and Ed Carr write about solutions to the horizontal line problem and testing of the fast A/D converter. A block diagram of Transformatrix with an account of how it will perform a parallel, two dimensional Fourier Transform on random pulse sequences, as well as details of the system's input mechanisms, come from Larry Ryan, Orin Marvel and Yiu Wo. Larry Wallman discusses a fundamental change in the design of the lenticular screen for the 3D Television project. Finally, the status of the system design for Stereomatrix is reported on by Shiv Verma and Richard Cheng.

M. Faiman

2.1 OLFT (Project No. 12)

2.1.1 Pattern Resolution

A first - order perturbation analysis of the light - valve modulation characteristics has been completed and will be issued as Report No. 303. One of the results of this analysis indicates that pattern resolution should be rotationally anisotropic (relative to the polarization of the incident light). Thus, for example, if a bar pattern is written on the crystal, the visible intensity or contrast of the resulting picture is dependent on the direction of the bars. This result is a consequence of the biaxial characteristics of the linear electro-optic effect in $\bar{4}2m$ crystals.

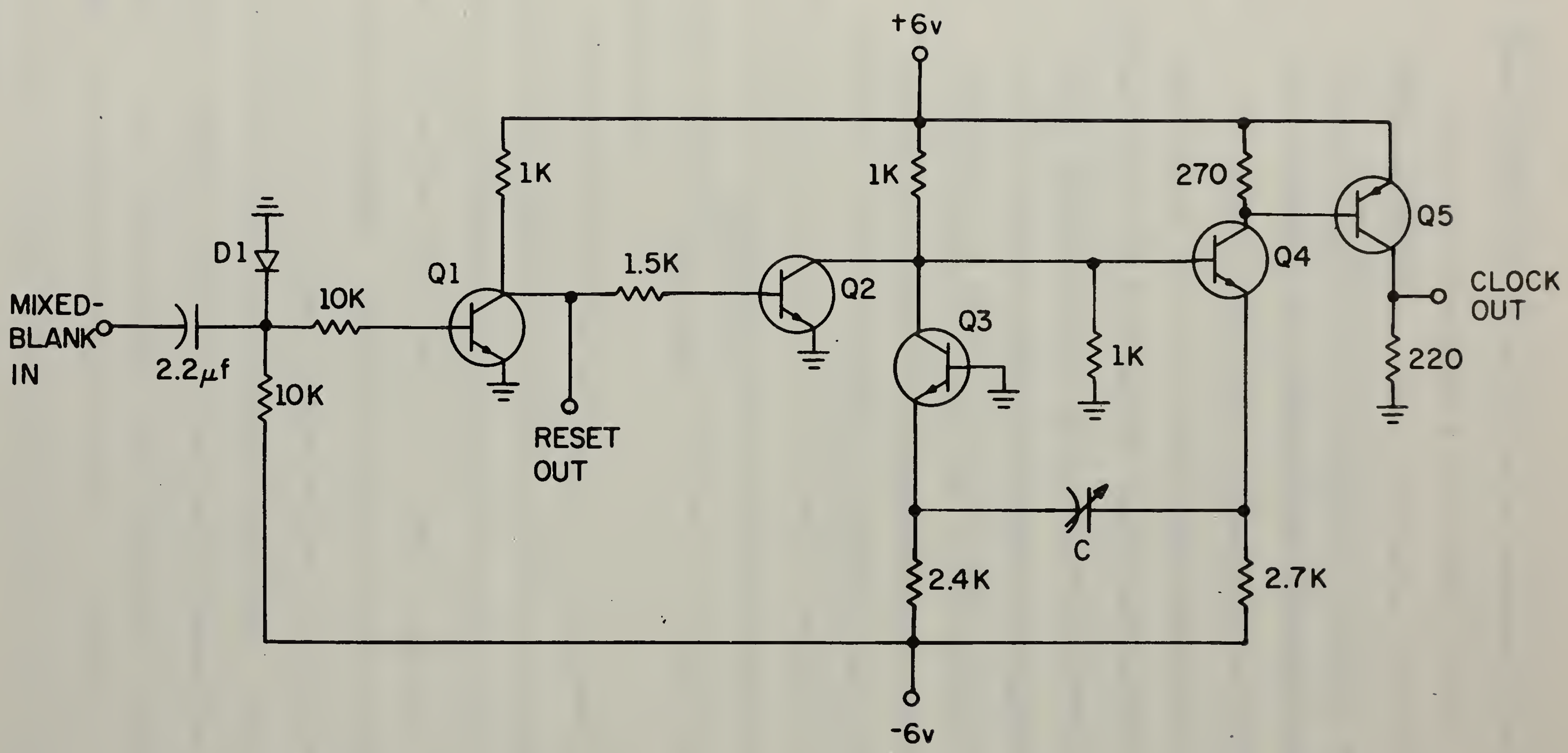
This first - order approximation indicates that the resolution anisotropy should be significant in "strength", suggesting that the resolution in two dimensions might be considerably improved by electronic "pre-emphasis" (simple high-pass filtering) along one dimension only.

To verify this theoretical analysis we have proposed a series of resolution tests using several available crystals. The selected pattern is a television raster containing uniformly spaced vertical bars. This pattern is not ideal but is simple to generate and easy to control and measure. The bars are square waves of variable spacing and are generated by a binary count-down from an astable multivibrator gated by the horizontal blanking pulse of the raster. The result is a pattern of constant "density" over all spatial frequencies, which is superior to the variable density of the traditional "shrinking-raster" method. To determine the angular variation in resolution, the pattern will be rotated by simply rotating the deflection coils around the electron gun.

The circuit which generates the bar pattern is shown in Figures 1 and 2. The gated "clock" (Figure 1) is a simple emitter-coupled astable multivibrator which generates a stable square wave at about 10 MHz. The clock is gated and phase-locked by the mixed blanking signal from a master television sync generator, and produces ~512 bars per horizontal line. The remainder of the bar generator is shown in Figure 2. The clock signal drives an 8 - bit binary counter which is reset by the blanking signal. A nine - position switch selects the desired number of bars per line and the resulting digital signal is converted to a clean square wave by transistors Q6 and Q7. The cable driver (Q8) completes the bar generator. The output of the bar generator does not contain a composite sync signal, so that sync must be supplied from the master sync generator.

Dave Casasent

Doug Sand



Q1-Q4: 2N709
Q5: 2N2894A
D1: 1N4151

(C ~ 60pf FOR ~9.5 MHz OR
512 BARS PER HORIZONTAL LINE)

Figure 1. Gated Clock

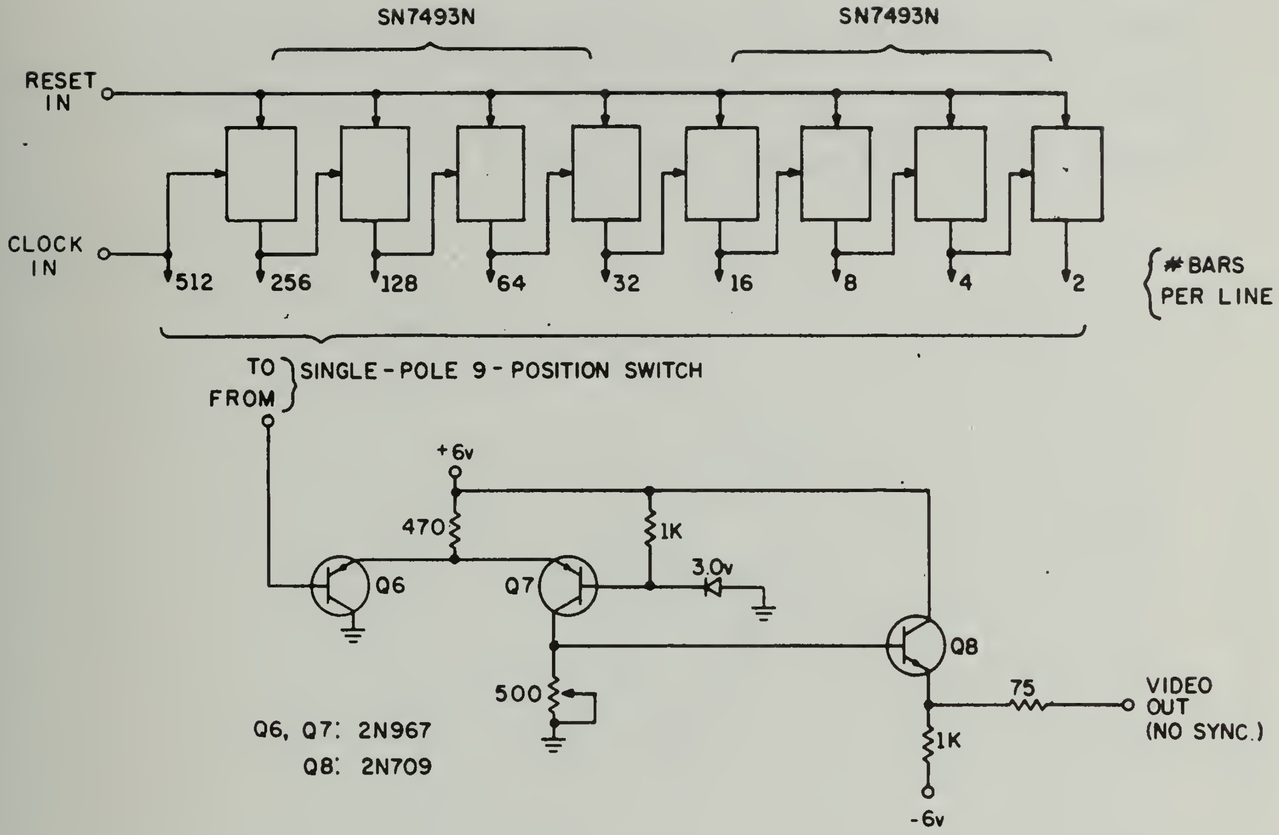


Figure 2. Bar Generator

2.1.2 Sync Separator Modification

The sync separator circuit discussed in a previous Quarterly Report proved to be too sensitive to variations in the level of the video signal. To correct this the sync-clipping stage was modified (see Figure 3) to include a peak detector (Q12) which gives stable sync-clipping for all video signal levels from 0.2 V to 4 V.

Al Irwin

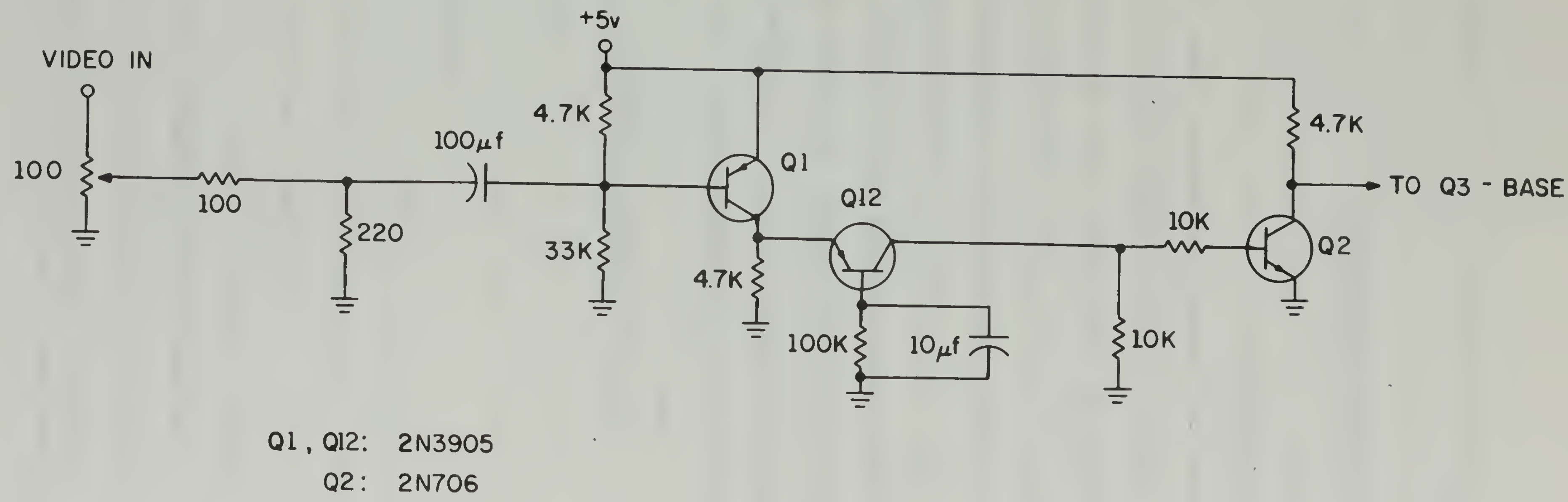


Figure 3. Sync - Clipping Modification

2.2 VISTA System (Project No. 14)

2.2.1 Variable Scan Television Camera

2.2.1.1 Video Preamplifier

To accommodate the 64 possible scanning rates, a variable gain video preamplifier is required since the signal level is, to a first approximation, a linear function of the scan rate*. 64 different values of gain would be prohibitively complicated and costly, hence nine different ranges of scan rates were selected, each corresponding to a different value of gain. Table 1 presents the nine selected ranges. n is the factor by which the conventional scan rate, 15750 Hz, is reduced. These ranges were chosen mathematically to minimize the error in the ranges which results from providing nine instead of 64 gain values. The maximum gain error is approximately ± 21%.

Range	Scan Rate Reduction Factor n	Gain Factor
1	1	1
2	2	2
3	3 - 4	3.5
4	5 - 7	6
5	8 - 11	9.5
6	12 - 17	14.5
7	18 - 26	22
8	27 - 41	34
9	42 - 64	53

Table 1. Scan Rate Ranges.

*RCA Review R. E. Johnson, "Vidicon Performance Characteristics at Slow Scan Rates", March 1966, Vol. XXVll, No. 1, Pages 61 - 65.

Figure 1 is a simplified diagram of the Vidicon-preamplifier circuitry. The gain is adjusted by varying the value of R_V, rather than by controlling the gain in later stages of the preamplifier. This technique provides a control of the noise level by reducing the high frequency response as R_V is increased for lower scan rates. The majority of the noise originates as thermal noise in R_V. By increasing R_V, the R_VC time constant is also increased, where C is the capacitance between the target mesh and all other electrodes in the vidicon. This decrease in frequency response offsets the increasing noise level which varies as the square root of R_V.

The input resistance R_V is controlled by using a field effect transistor as a voltage controlled resistor. The voltage, in turn, is controlled by a resistor divider composed of a standard resistor and a Hewlett-Packard photocell-lamp combination. This arrangement offers excellent isolation between the controlling mechanism and the input resistance R_V, an important consideration at such low signal currents. This circuit is shown in Figure 2. A feedback signal is returned to the gain control circuit from the preamplifier in order to bootstrap the gate and null the effect of the gate-to-drain capacitance. This technique is discussed in more detail below where bootstrapping is applied to the preamplifier.

A field effect transistor was used at the input of the preamplifier for its high input impedance characteristics. This prevents attenuation of the already small signal level. Figure 3 shows the preamplifier circuit. Here Q_2, used as an emitter follower, bootstraps the source and drain through C_1 and C_2, respectively. An analysis of the circuit shows that such an arrangement eliminates the effect of

the gate to source and drain to source capacitances provided that the emmitter follower voltage gain is unity. In a similar manner, the collector of Q_2 is bootstrapped through C_3. The input capacitance to the preamplifier circuit was measured at approximately 0.8 pf. The input resistance is well over 10 Megohms. A combination of series and shunt peaking is employed in stages Q_5 and Q_6.

In order to enhance television picture detail, conventional television preamplifiers employ a peaking stage to boost the high frequency response. This technique was employed in this preamplifier design, though not to the extent that it nulled the noise reduction effect described above. Here an additional complexity arose due to the variable gain feature of the preamplifier. As the upper frequency limit falls off with increasing R_v, as discussed above, the peak must shift accordingly. Thus a different pole-zero combination is switched into the circuit for each of the nine gain settings. This is shown in the emitter circuit of Q_7 in Figure 3. The networks switch simultaneously with the nine different values of R_v. The frequency response is thus flat to the nine different cutoff points. The other eight bandwidths are reduced from this figure by the gain factors listed in Table 1.

For a gain factor of 1, the voltage gain of the preamplifier is approximately 40 db.

2.2.1.2 Deflection Amplifiers

Due to the shifting nature of the television raster, direct-coupled deflection amplifiers are required to drive the plates of the electrostatic vidicon tube. The specifications called for 30 volts peak-to-peak deflection voltage per plate with an average voltage of 225 volts. An inverting and a non-inverting amplifier circuit were

designed using operational amplifiers and high voltage transistors as shown in Figure 4. The signal outputs are returned through feedback resistors to the inputs. One such pair of amplifiers is used to drive the horizontal plates, and an identical pair to drive the vertical plates. The linearity and response of the amplifiers are excellent over the entire range of scan rates.

2.2.1.3 Vidicon Protection Circuits

A circuit for protection against horizontal deflection failure was designed consisting essentially of a differentiating and threshold detector circuit. If the slope of the ramp deflection voltage falls below a certain minimum value (corresponding to the slowest scan rate), the derivative of this signal, a step voltage, will fall below the corresponding threshold voltage, triggering the blanking circuit and removing the target voltage. A different scheme was required for the stair-step vertical deflection waveform. In this case, the voltage change between steps is monitored at the conclusion of every horizontal sweep. If no change occurs, the vidicon blanking circuit is activated.

2.2.1.4 Range Discriminator Circuit

A range discriminator is required to decode the 64 possible scan rates into nine different ranges. Refer to Table 1. In the first two cases where there is a one-to-one correspondence between scan rate and gain, a digital decode is used. In the following cases, however, an analog approach is much less complex and costly. Here the digital signal corresponding to the selected scan rate is converted to an analog voltage. Using dual differential amplifier packages, a window corresponding to each digital range is selected. The output of each amplifier pair thus represents one of the nine different

ranges. A simplified circuit diagram of this arrangement is shown in Figure 5. The inverted outputs of the amplifier pairs drive the respective L'C' switches shown in Figure 3. In addition, these outputs drive a lamp intensity control circuit for controlling the gain as shown in Figure 2. An attractive feature of this scheme is that the integral nature of the scan rates allows the windows to be chosen such that no overlap occurs, and the scan rate numbers will always fall within one and only one window.

2.2.2 Variable Cutoff Low Pass Filter

An adjustable low pass filter will be inserted in the video line to determine the bandwidth of the video signal and to demonstrate picture degradation as the width of the pass band is reduced. A sixth order active filter was designed consisting of three second order stages cascaded. The parameters were chosen to provide a max flat filter, i.e. the complex conjugate poles were chosen to lie on the unit circle, separated from each other by thiry degrees. Refer to Figure 6. The amplifier gains A_1, A_2, A_3, represent additional degrees of freedom. These values were kept within practical limits and were chosen to render the system stable.

Figure 7 shows the 10 MHz amplifier circuits used in stages 1 and 3. Stage two was a simple emitter follower-attenuator circuit and is not shown.

The cutoff frequency is adjustable over four ranges, spanning the limits of 16 KHz and 10 MHz. The sixth order system provides a cutoff slope of approximately 36 db per octave. The experimental gain versus frequency plots correlated extremely well with the theoretical plots.

2.2.3 Miscellaneous

A six-bit binary-to-binary coded decimal converter was designed and built to drive the Nixie tube bandwidth compression ratio display. In addition, camera power supplies and other miscellaneous supplies have been designed and built.

The camera is currently being built and assembled and will soon be operational. The variable persistence monitor has been received and installed in the system cabinet. The control panel is about to be wired.

David Rollenhagen

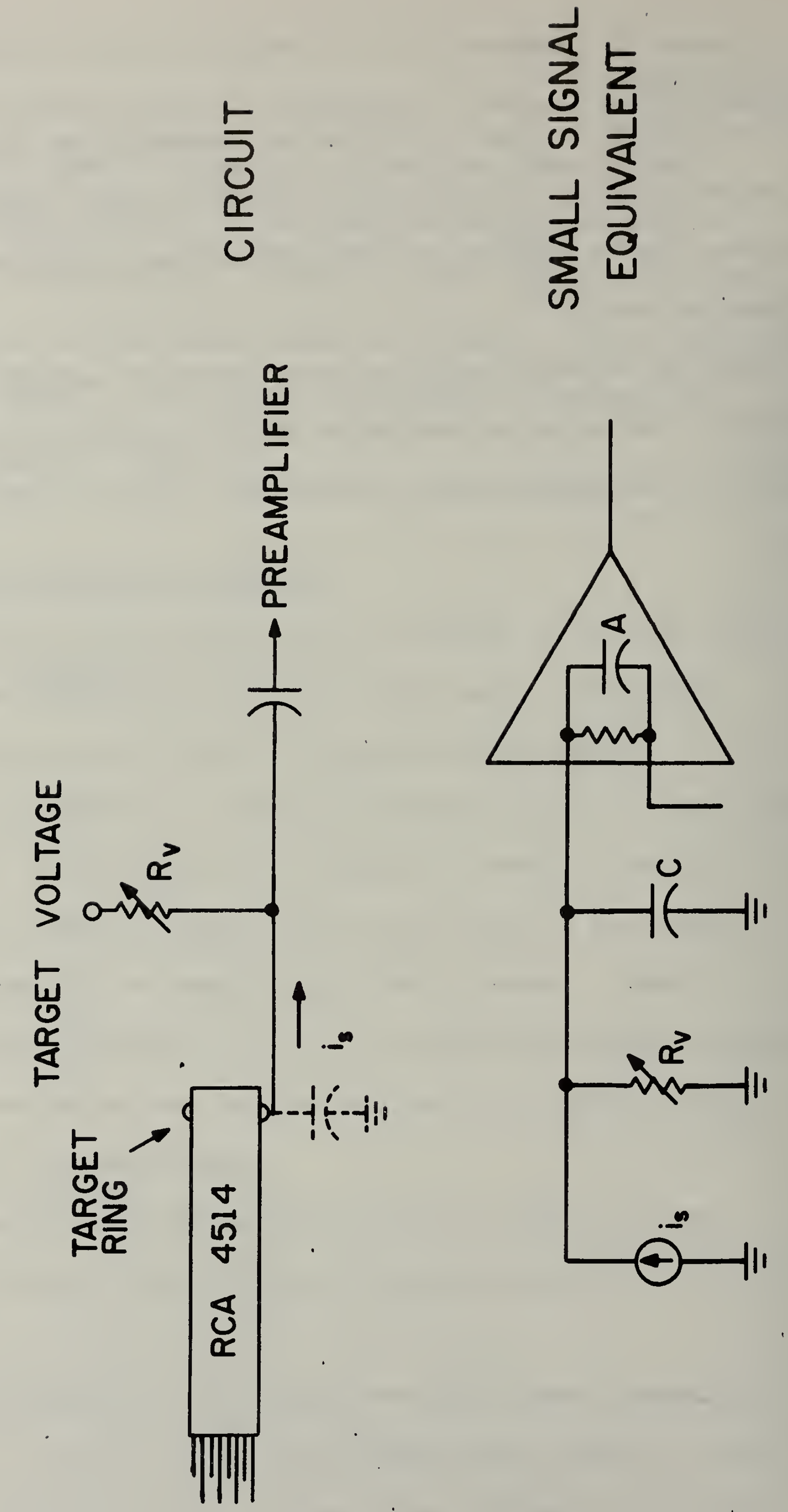


Figure 1. Vidicon Circuit and Equivalent

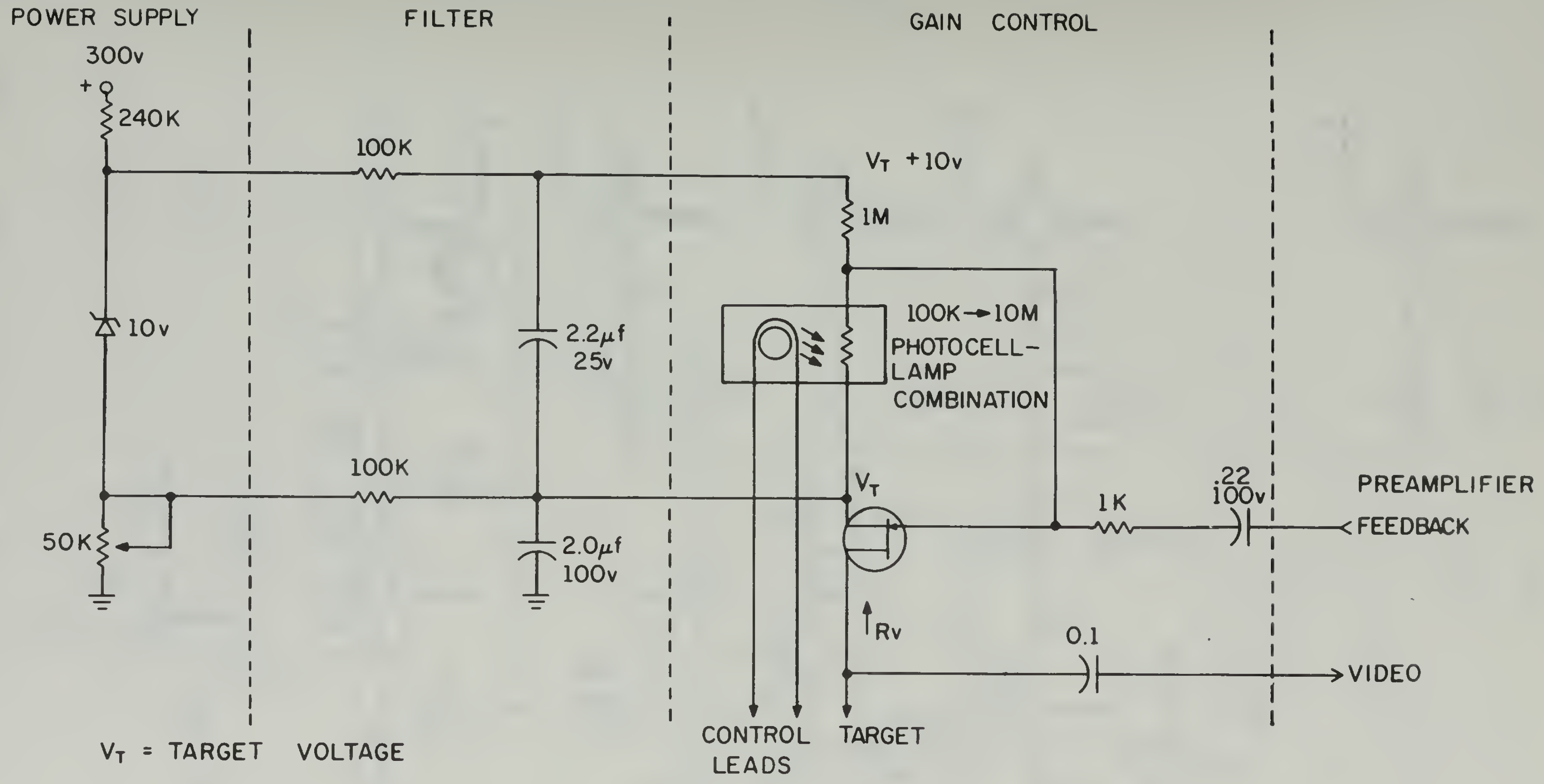


Figure 2. Gain Control Circuit

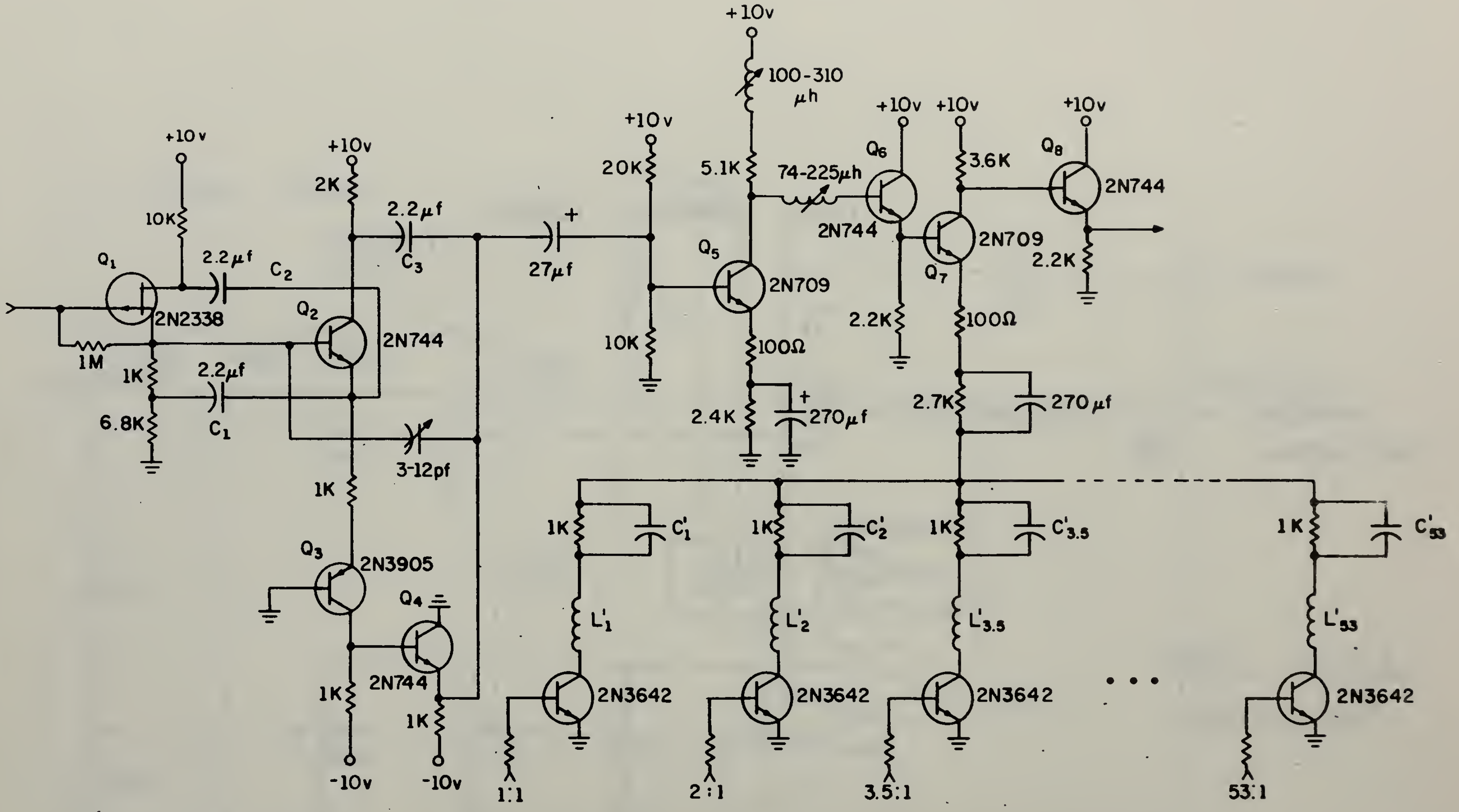


Figure 3. Preamplifier Circuit.

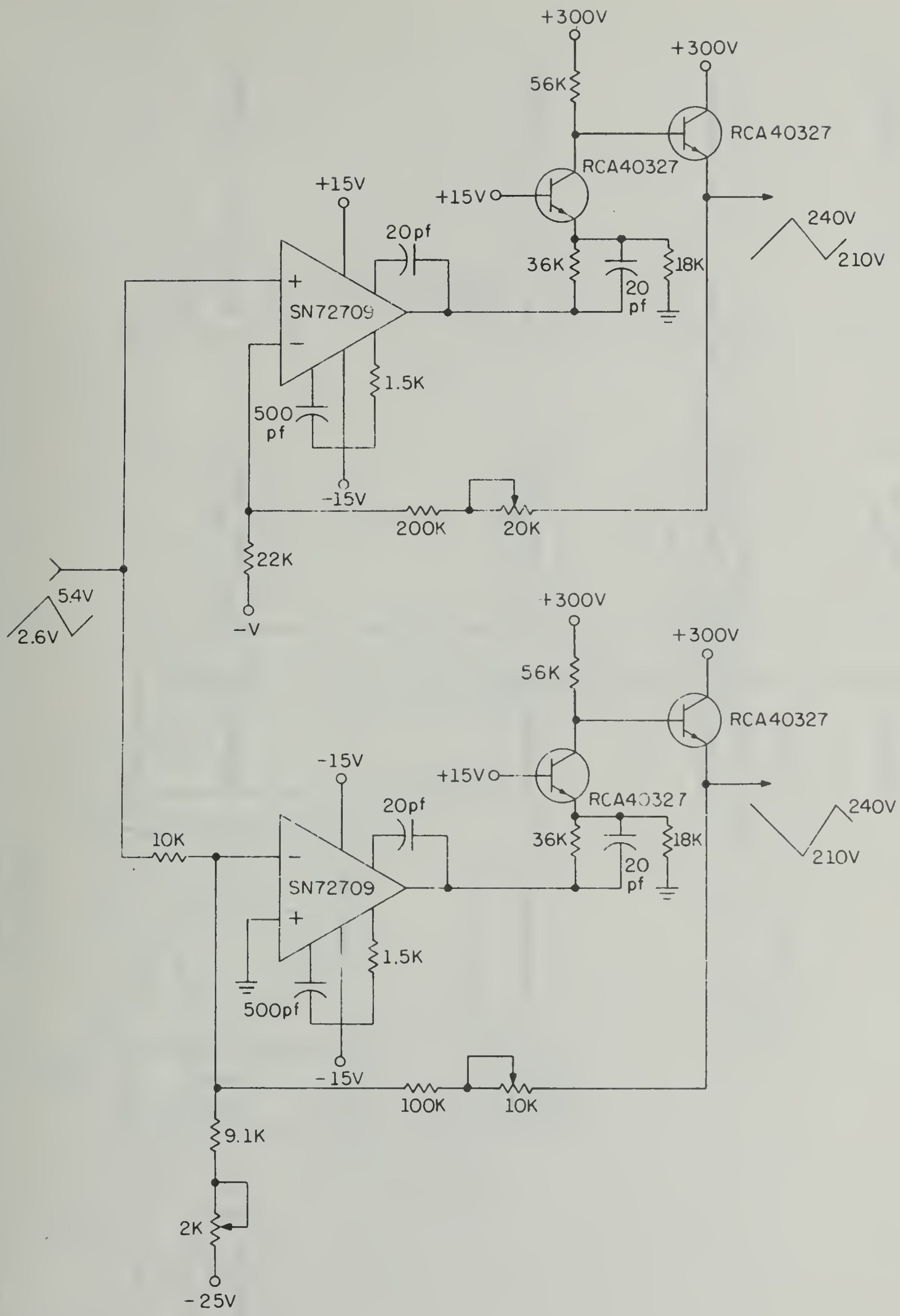


Figure 4. Direct Coupled Deflection Amplifier

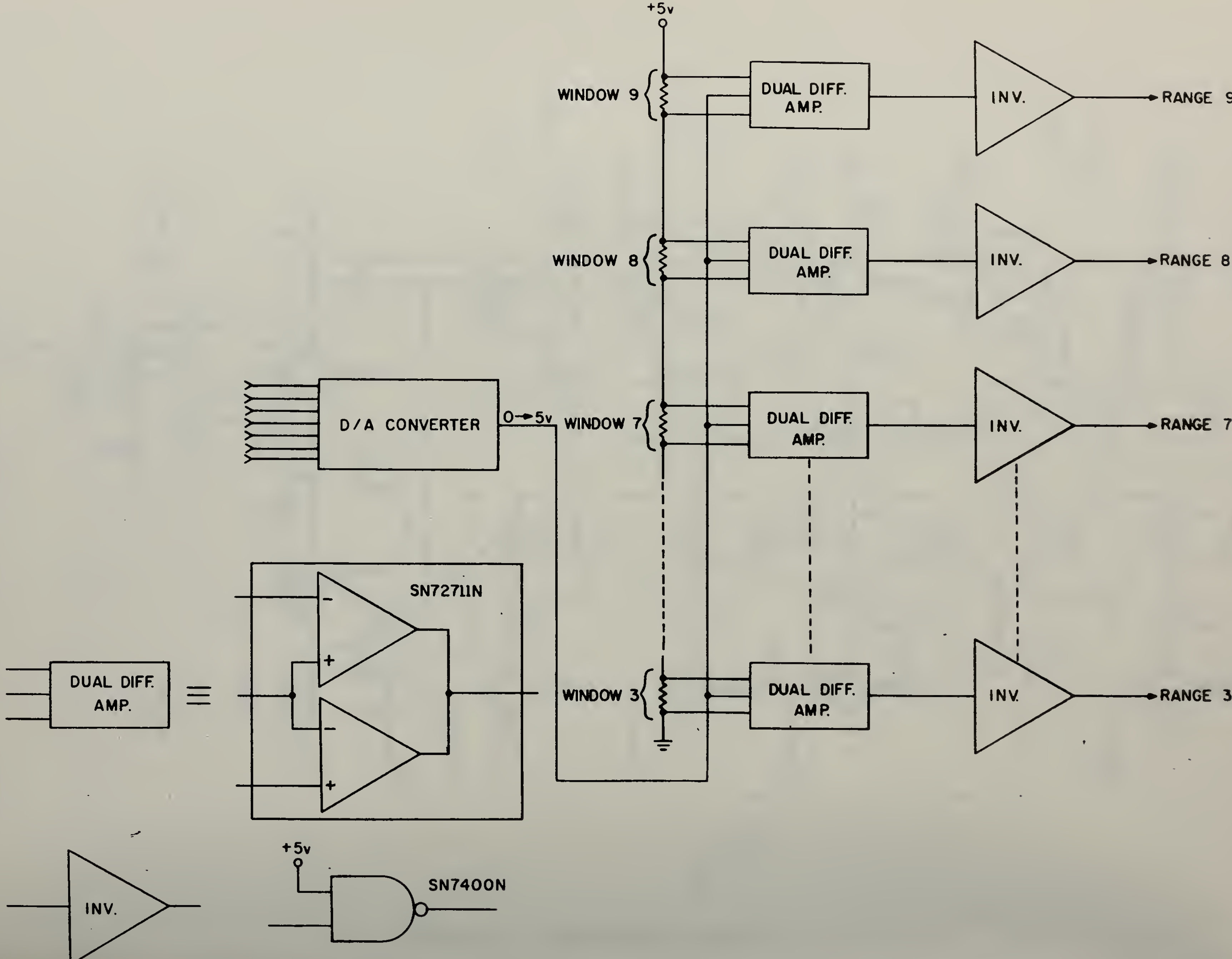
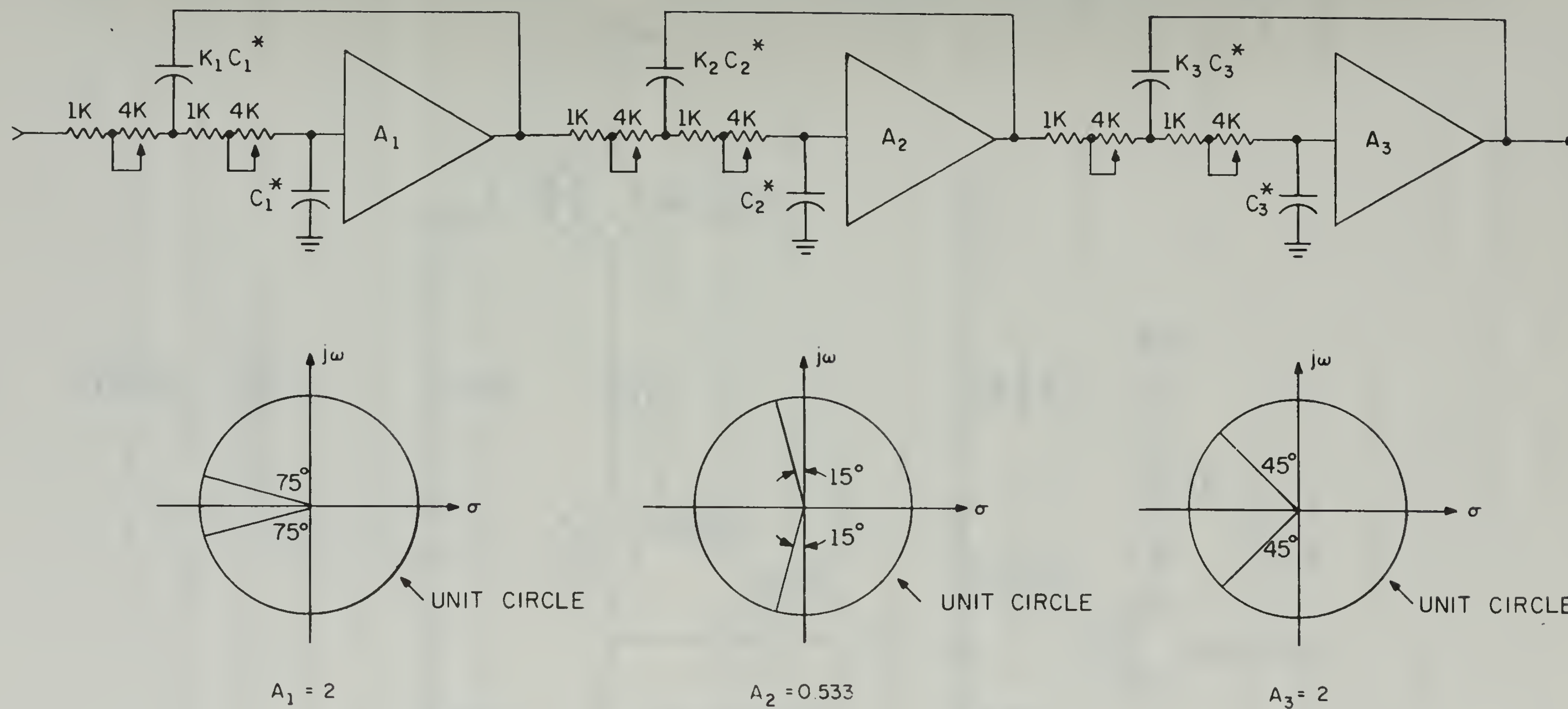


Figure 5. Range Discriminator.



* A ROTARY SWITCH SELECTS ONE OF FOUR DIFFERENT VALUES FOR $C_1, K_1C_1, C_2, K_2C_2, C_3, K_3C_3$ CORRESPONDING TO FOUR DIFFERENT RANGES. THE 4K POTENTIOMETER IS THE VERNIER.

Figure 6. Adjustable Sixth Order Low Pass Active Filter

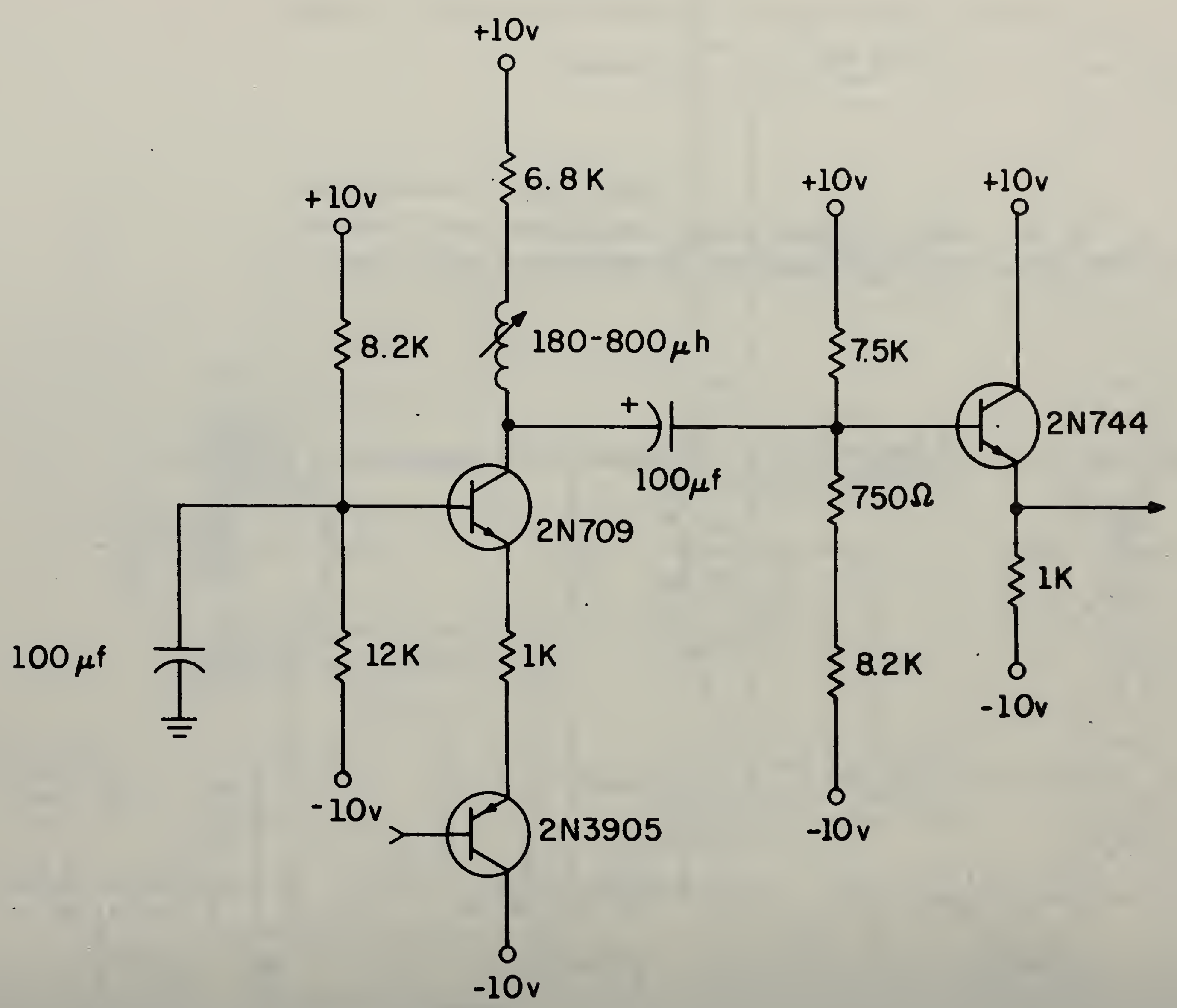


Figure 7. 10 MHz Amplifier with Gain of Two

2.3 Functional Encoding (Project No. 15)

As mentioned in an earlier report, the video signal from the camera is differentiated before detection, in order to get rid of the effect of the curvature of the background video level. An unfortunate consequence of this fact is that the intersection of the scan with a horizontal, or nearly horizontal, line to be transmitted produces too small a signal to appear at the output of the differentiator, so that information is lost.

Two solutions to the problem are being examined. One is to eliminate, electronically or optically, the curvature on the background video level. This would allow the differentiator to be removed and a simple threshold detector could then pick up the intersection of the scan with the line. The alternative, possibly less attractive, method is to alternate the scan horizontally and vertically from one frame to the next. Operating a vidicon in this mode does not seem too difficult, but the two dimensional alignment problem on the receiving monitor would not be so easy to overcome.

2.3.1 A/D Converter

The A/D converter previously described has been built and tested. Oscillations have been observed on one of the circuit boards containing a large number of high gain differential amplifiers which compare the input signal with reference voltages. Moderate success has been acheived in controlling the amplitude of the oscillations, but they have not yet been eliminated entirely. It looks at present as if greater physical isolation between input and output paths is needed, and a new layout is being tried.

Ed Carr

Peter Oberbeck

2.4 Transformatrix (Project No. 17)

2.4.1 Two Dimensional Discrete Fourier Transform

Let f (k,l) be the doubly periodic extension of a discrete input pattern, X_{kl}, such that:

$$f\,(k,l) = X_{kl} \quad , \qquad k,l = 0,1,\ \ldots,\ N-1 \qquad (1)$$

$$f\,(k + pN,\ l + qN) = X_{kl} \quad , \qquad \begin{array}{l} k,l = 0,1\ldots,\ N-1 \\ p,q \text{ integers} \end{array} \qquad (2)$$

We may then define a truncated Fourier series for the two dimensional discrete function f (k,l) in the following manner:

$$f\,(k,l) = \frac{1}{N}\sum_{m,n=0}^{N-1} F(m,n)\ \exp[2\pi i\ (km + ln)/N],\ k,l = 0,1\ldots,N-1 \qquad (3)$$

$$F\,(m,n) = \frac{1}{N}\sum_{k,l=0}^{N-1} f(k,l)\ \exp[-2\pi i\ (km + ln)/N],\ m,n = 0,1,\ldots,N-1 \qquad (4)$$

That expressions (3) and (4) do indeed form a Fourier Transform pair can be readily verified with the aid of the orthogonality relationships for the exponential function.

In addition to the Coordinate Transformation (QTPR - AMJ -1968), Transformatrix will be capable of realizing equation (4), a two dimensional discrete Fourier Transform, with N = 32. Since F(m,n) is in general a complex quantity, Transformatrix will display any one of the three functions $|F(m,n)|$, $|Re[F(m,n)]|$, $|Im[F(m,n)]|$.

We now consider some of the properties of the discrete Fourier Transform. Through direct use of (4) we see that:

$$F(m,n)^* = F(32-m,\ 32-n) \qquad (5)$$

Thus, for each output frame of F(m,n) values that are to be calculated, only $\frac{1}{2}[(32)^2 + 4] = 514$ uniques values need be determined with the other 510 values obtained through the use of (5). This reduces the

computation time by almost one-half.

Let the input picture be normalized such that $0 \leq f(k,l) \leq 1$. Then the range of values of $F(m,n)$ for $m = n = 0$ is 0 to 32. The range of values of $Re[F(m,n)]$ and $Im[F(m,n)]$ for non-zero m or n is 0 to $\pm$ 16. Care will have to be taken to preserve accuracy through this relatively large range of values for the points in the output picture.

From (4) we have for $Re[F(m,n)]$ and $Im[F(m,n)]$;

$$Re[F(m,n)] = \frac{1}{32} \sum_{k,l=0}^{31} f(k,l) \cos(2\pi(km + ln)/32) \qquad (6)$$

$$Im[F(m,n)] = - \frac{1}{32} \sum_{k,l=0}^{31} f(k,l) \sin(2\pi(km + ln)/32) \qquad (7)$$

In order to form $|F(m,n)|$, $|Re[F(m,n)]|$, and $|Im[F(m,n)]|$ we represent $f(k,l)$, $\cos(2\pi(km + ln)/32)$, and $-\sin(2\pi(km + ln)/32)$ as Synchronous Random Pulse Sequences (SRPS) and use digital AND's to obtain the products in (6) and (7). The $f(k,l)$ values are already in the proper form $(0 \leq f(k,l) \leq 1)$ to be represented by SRPS. However, the $\cos(2\pi(km + ln)/32)$ and $-\sin(2\pi(km + ln)/32)$ values range from -1 to +1 and a mapping is required. The type of mapping that will be used is:

$$P_{cos}(km + ln) = \frac{1}{2}(1 + \cos(2\pi(km + ln)/32) \qquad (8)$$

$$\text{where } 0 \leq P_{cos}(km + ln) \leq 1 \qquad (9)$$

$P_{sin}(km + ln)$ will be obtained by incrementing the argument of $\cos(2\pi(km\ 2\ ln)/32$ by $\pi/2$.

$$P_{sin}(km + ln) = \frac{1}{2}(1 + \cos(2\pi(km + lm + 8)/32 \qquad (10)$$

$$\text{where } 0 \leq P_{sin}(km + ln) \leq +1 \qquad (11)$$

Multiplying the SPRS which represents $F(k,l)$ by the SRPS's which represent $P_{cos}(km + ln)$ and $P_{sin}(km + ln)$ and summing (in an analog manner) over all (k,l) we obtain:

$$Re[F'(m,n)] = \frac{1}{32} \sum_{k,l=0}^{31} f(k,l)[\frac{1}{2}(1 + \sin(2\pi(km + ln)/32))]$$

$$= \frac{1}{2} F(0,0) + \frac{1}{2} Re[F(m,n)] \qquad (12)$$

$$\mathrm{Im}[F'(m,n)] = \frac{1}{32} \sum_{k,l=0}^{31} f(k,l)[\tfrac{1}{2}(1 - \sin(2\pi(km + ln)/32))]$$

$$= \frac{1}{2} F(0,0) + \frac{1}{2} \mathrm{Im}[F(m,n)] \tag{13}$$

The quantity F(0,0) represents the average intensity of the input picture. Thus, for each (m,n) we have the result:

$$\mathrm{Re}[F(m,n)] = 2\mathrm{Re}[F'(m,n)] - F(0,0) \tag{14}$$

$$\mathrm{Im}[F(m,n)] = 2\mathrm{Im}[F'(m,n)] - F(0,0) \tag{15}$$

Reviewing the preceding equations it is seen that the crux of the problem is the generation of the coefficients, $P_{cos}(km + ln)$ and $P_{sin}(km + ln)$, as SPRS's. Several approaches for the generation of the (Fourier) coefficients were discussed in the QTPR-AMJ-1968. A block diagram of the scheme which was considered superior is found in Figure 1. The number in parentheses above each box indicates the number of identical circuits represented by that box.

As indicated in Figure 1, there is one five bit adder for each column of coefficients. Because of the periodicity of the cosine and sine functions, we are only interested in (km + ln)mod 32.

For a particular output point and a fixed column of coefficients, the indices m, n, and k are constant. Assume that $(km)_{32}$ and 0 are entered into the input registers of the k^{th} adder. The decoder converts $(km + ln)_{32}$ into $P_{cos}(km + ln)$. Thus $P_{cos}(km)$ is entered into the l = 0 five bit register of column k. Then the number $(km)_{32}$ is fed back into one of the input registers of the adder. The other register now contains n. Thus the output of the adder becomes $(km + n)_{32}$ The decoder generates $P_{cos}(km + n)_{32}$ which is entered into the 'l' register of column k. This process continues (with one input to the adder equal to n and the other input equal to the previous output) up to l = 31 for all 32 columns. The resulting five bit binary numbers stored in the 1,024 registers feed 1,024 D/A's. The output of each

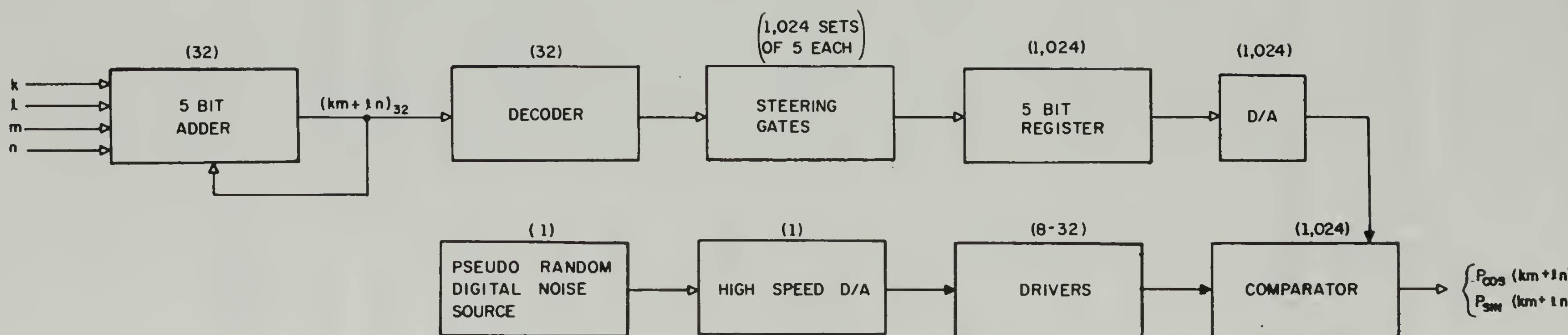


Figure 1. Block diagram for the generation of Fourier coefficients as SRPS.

D/A is compared with a random analog voltage which is generated from a SRPS. The output of each comparator is a SRPS representing $P_{cos}(km + ln)$. Recall that $P_{sin}(km + ln)$ is obtained by initializing one of the input registers of the adders by a binary 8.

$$P_{sin}(km + ln) = P_{cos}(km + ln + 8) \tag{16}$$

Details of the individual circuits will be presented in subsequent reports.

2.4.2 Transformatrix Inputs

Two input mechanisms will be attached to Transformatrix: a slide projector, for testing and aligning circuitry and for demonstrating the coordinate and Fourier transformation properties of the system; and a 5" TV monitor with provision for line camera input, for demonstrating the dynamic capability of Transformatrix. The optical information from either of these devices is projected on to a square array of 1024 photoconductors, as shown in Figure 2. Circuits connected to the photoconductors (currently being tested) convert the sampled image points into SRPS's.

The chief reason for using a photoconductive matrix as an input transducer is that it solves, in an automatic fashion, the problem of converting the sequential nature of the TV display to a parallel one. A photoconductor responds asymmetrically to the leading and trailing edges of a light pulse, turning on - i.e. going into high conductivity- much faster than turning off. It is possible to obtain such devices with fall times long enough to "store" the (optical) information for one TV frame time. Care must be excercised, however, in not making the decay time so long as to blur the effect of rapidly changing images. Preliminary calculations indicate that a decay time of the order of 0.3 second will be commensurate with 16 gray levels of illumination in the range from about 0.1 to 1.0 foot-candle.

L. Ryan
O. Marvel
Y. Wo

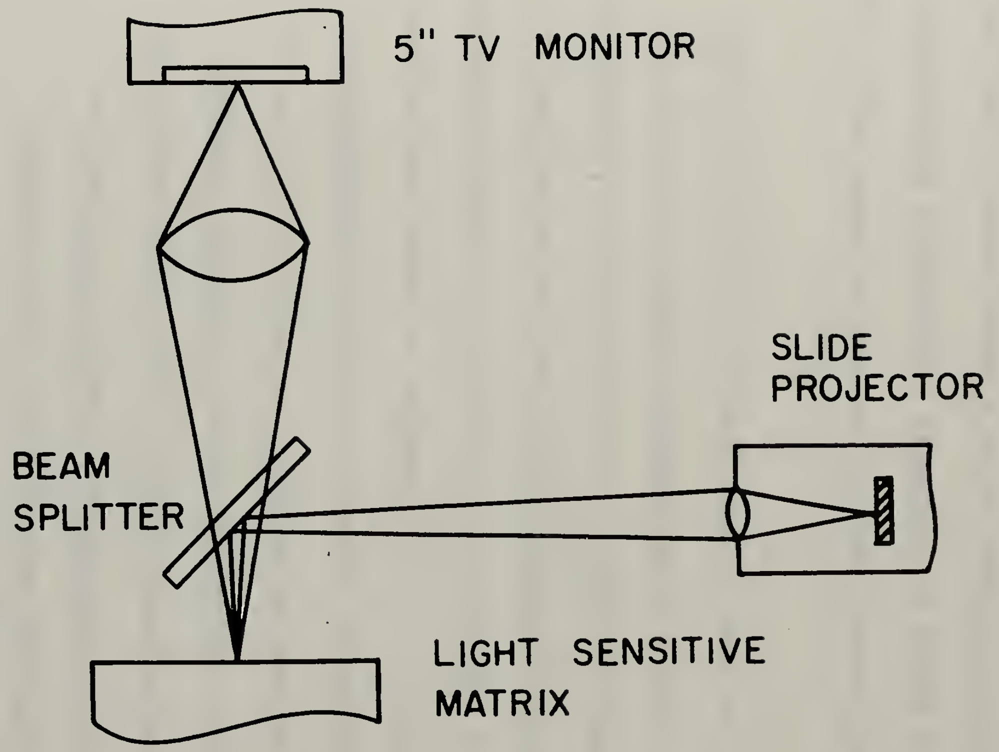


Figure 2. Inputs to Transformatrix

2.5 3D Television (Project No. 19)

2.5.1 Special CRT

It has been decided to abandon the vertical scan system with the fibre optic faceplate and to pursue a system with a horizontal scan and an internal lenticular screen. The horizontal scan method can be used if the video signal applied to the monitor is composed of the video from two television cameras which has been chopped and mixed so that alternate left and right video information appears under each lenslet on the lenticular screen.

In as much as a special CRT is needed, suppliers have been contacted and several have been found who can make the tube, but none is willing to make the lenticular screen. The lenticular glass screen is to be 6" by 8" by approximately 3/32" and have 1/32" radius cylindrical lenses pressed into one side. This apparently can be done by machining a ceramic form and pressing the lenses in under heat. Ceramic material has been ordered to make the form and the work will be done in a shop at the University of Illinois.

The lenticular screen will have conducting stripes deposited on it, so that a feedback signal can be obtained from the electron beam as it scans them. This feedback will be used to synchronize the chopper-mixer so that the left-right information appears in the proper places under the lenses.

Specifications for the television cameras have also been written and quotes asked for.

L. H. Wallman

2.6 Stereomatrix (Project No. 30)

2.6.1 System Design

The general scheme of Stereomatrix has already been presented in the previous report. It has been decided to use fast hybrid techniques for various functional units in the system. A block diagram for the system has been drawn up and critical elements such as hybrid multipliers and dividers have been investigated and ordered for laboratory evaluation.

As shown in Figure 1, the picture coordinates (triplets) are stored on a disc in the frame (x^0, y^0, z^0). The section of this picture being displayed is stored in display memory, which is referred to as the display volume, and has coordinates (x^1, y^1, z^1). These are each of 10 bits. They are transformed into the viewing screen frame (x^2, y^2, z^2) by the Coordinate Transformer (Figure 2).

The transformed coordinates in the viewing screen can be rotated about the x, y, or z axes, as desired by the viewer. However, before rotation is performed a scaling unit scales down the volume for ease in displaying it. The rotation can be performed either by turning a set of knobs provided on a control panel or rotational command obtained from the disc.

The rotated coordinates are used to generate a stero pair for each incoming coordinate. However, these stereo pairs are only displayed when they lie in the window area of the display; otherwise they are rejected and a new set of data is requested. The windowing unit determines whether the obtained pair is in the window area or not and gives its decision to the Display Control unit. The latter controls the input data, bypasses the rotational unit if no rotation is desired and requests new coordinates if the old ones are out of the window area.

The intensity control unit provides the intensity signal to the

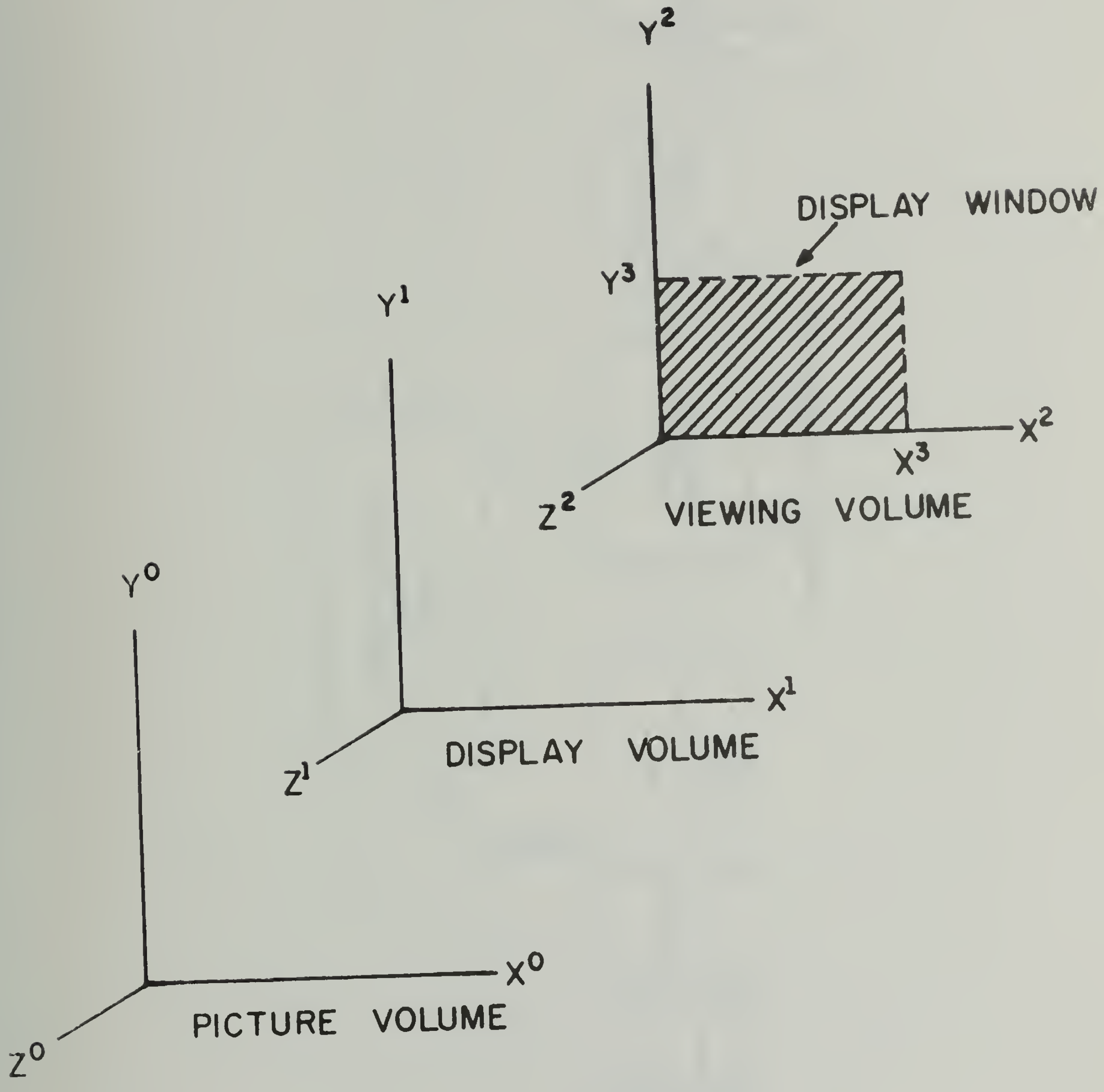


Figure 1. The Various Reference Frames of Stereomatrix

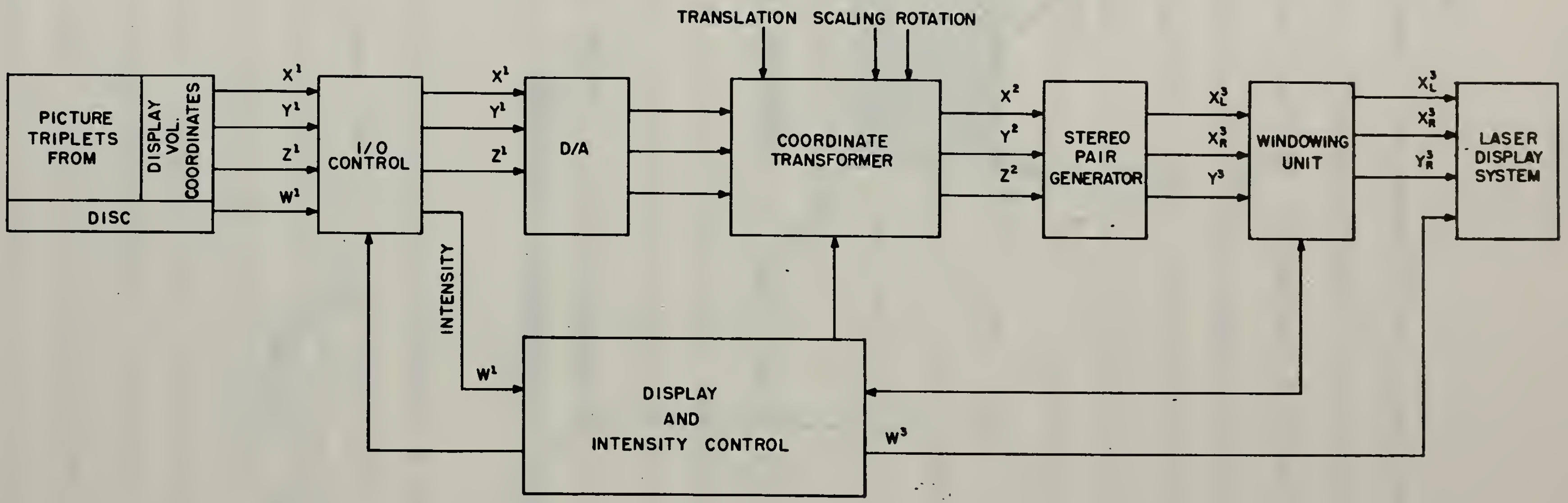


Figure 2. Block Diagram of Stereomatrix

laser display system if the stereo-pair lies within the window area.

Figure 3 shows the generation of the stereo-pair.

The problem which remains to be solved is a simple but adequate scheme for rotation of the picture being displayed.

Shiv Verma

Richard Cheng

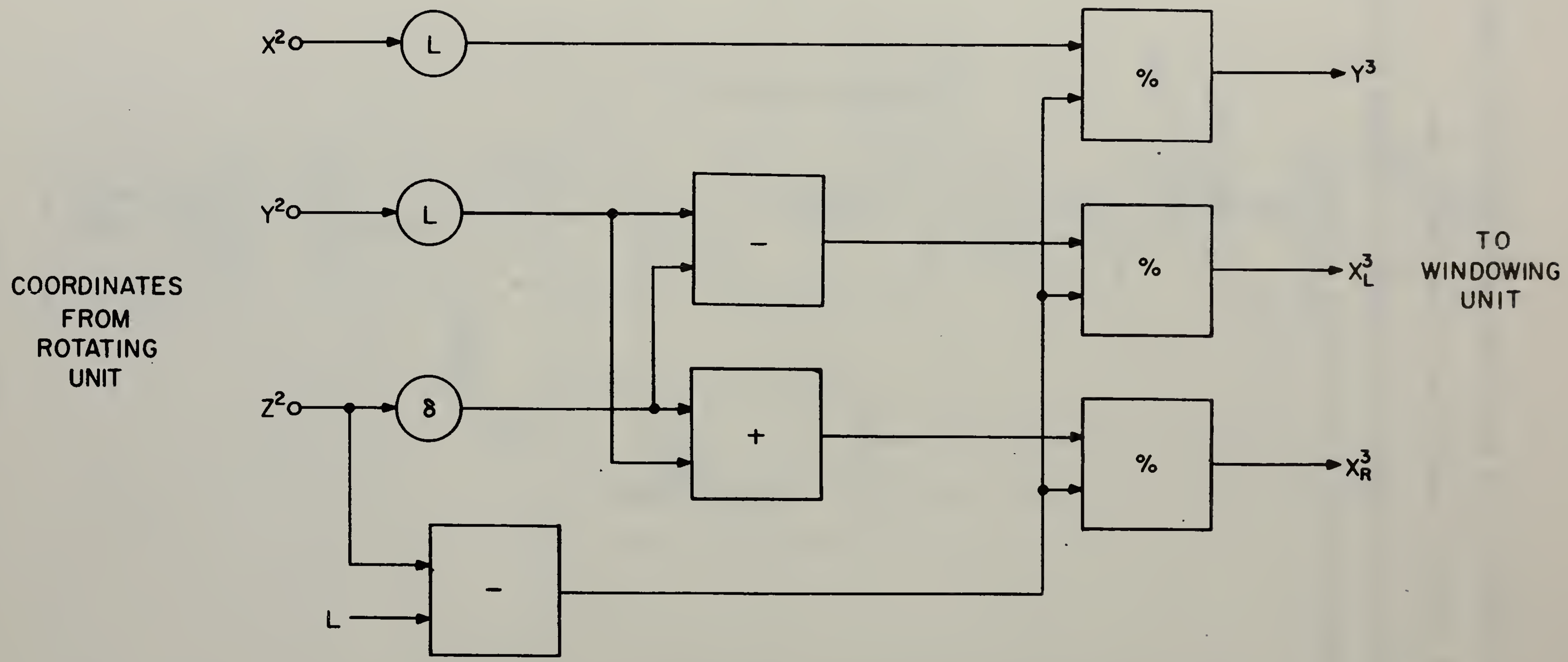


Figure 3. Block Diagram of Coordinate Transformer

3. Computer Systems Research

Supported by the U. S. Atomic Energy Commission under grant U. S. AEC AT(11-1)1469, Part A. The work performed for this quarter is reported in the next six subsections. Publications and reports prepared or published this quarter include:

Reports Written

1. Dill, C. "A Graphical Search for Stiffly Stable Methods for Ordinary Differential Equations," Department of Computer Science Report No. 295 and Master's Thesis, University of Illinois, Urbana, Illinois 61801.

2. Dill, C., Ellis, C., Gear, C. and Ratliff, K., "The Automatic Integration Package for Ordinary Differential Equations," Department of Computer Science File No. 779, University of Illinois, Urbana, Illinois 61801.

3. Dill, C., Ellis, C., Gear, C. and Ratliff, K. "ODESSY--Ordinary Differential Equation Solver System," submitted to Communications of the ACM.

4. Dill, C. and Gear, C. "A Graphical Search for Stiffly Stable Methods for Ordinary Differential Equations," submitted to Journal of the ACM.

5. Gear, C. "General Simulation and Modeling," Department of Computer Science File No. 785, University of Illinois, Urbana, Illinois 61801.

Papers Published

6. Gear, C. "The Automatic Integration of Stiff Ordinary Differential Equations," Proceedings IFIPS 1968, pp. A81-A85.

7. Richardson, F. K., Lo, T. Y. and Gear, C. W. "Computer Aided Programming System," in "Emerging Concepts in Computer Graphics," ed. Secrest and Nievergelt, W. A. Benjamin Book Company (1968), pp. 171-188.

Papers Presented Orally

8. Gear, C. "The Automatic Solution of Ordinary Differential Equations," presented to the Symposium on Mathematics, Computers and Computation, Oct. 1968, Urbana, Illinois.

C. W. Gear, Principal Investigator

3.1 Numerical Analysis--Ordinary Differential Equations

Considerable success has been enjoyed this quarter in this work. New stiff methods were found by a graphical search (Section 3.1.1) and considerable use of the techniques and programs has been made by other groups around the world. The basic numerical section is in use at Argonne National Laboratory and the Danish Atomic Energy Laboratory to solve reaction problems. The method is now being used in an electrical network analysis program by Calahan of the University of Michigan. A joint paper is in preparation. Part of the quarter was spent on applying the method to a complex set of equations obtained from the Kirkland group. A speed in excess of 100 was obtained with better accuracy. As a result of this, and of Calahan's tests, a number of other groups have sent for the program to test it on their problems.

All known remaining errors in the automatic package have been corrected and a letter has been sent out announcing the availability of tape copies of the program.

3.1.1 An Interactive Search for Stiff Methods (C. Dill)

An interactive graphic system for displaying stability functions useful in connection with the study of high order methods for solving stiff ordinary differential equations is now operational on the PDP-8/338. Known stiffly stable methods are predictor-corrector methods. The corrector may be written

$$\alpha_k y_n + \alpha_{k-1} y_{n-1} + \ldots + \alpha_0 y_{n-k}$$

$$+ h\xi \; \beta_k y'_n + \beta_{k-1} y'_{n-1} + \ldots + \beta_0 y'_{n-k} \xi = 0$$

where $\alpha_k \neq 0$, $|\alpha_0| + |\beta_0| > 0$. Define

$$\rho(\xi) = \sum_{i=0}^{k} \alpha_i \xi^i \text{ and } \sigma(\xi) = \sum_{i=0}^{k} \beta_i \xi^i .$$

The stability function displayed is the locus of $-\frac{\rho(\xi)}{\sigma(\xi)}$ in the complex plane, with $\xi = e^{i\theta}$.

Given a k and the coefficients of the polynomial $\sigma(\xi)$, the coefficients of the polynomial $\rho(\xi)$ and thus the unqiue method of degree k are completely determined. The stiff stability properties of the method may be investigated by a visual examination of the displayed locus. The display is based on a user defined display window specified in terms of left and right boundaries relative to the real axis, and upper and lower boundaries relative to the imaginary axis. Communication to the system is by teletype and push-buttons. User input includes k, the coefficients of the polynomial $\sigma(\xi)$, and the display window boundaries.

Previous results showed the existence of stiffly stable methods of order up to six. The graphic system has been used to conduct a systematic search for methods of order greater than six. This search has led to the discovery of stiffly stable methods of orders seven and eight.

Defining optimality in terms of least amount of solution history data to be saved at each step of the corrector iteration, the optimal possibilities for orders seven through fifteen were tested with negative results for all orders except seven. Removing the optimality restriction allows the unique method determined by any stable $\sigma(\xi)$ to be a valid candidate for satisfying stiff stability requirements. No general results can be given, since the number of potential methods excludes the possibility of any type of exhaustive testing. Testing of large classes of methods tends to indicate the scarcity of stiffly stable methods of order greater than six. Methods of orders seven and eight were found, however, while examining some of the more likely classes of methods.

3.1.2 Automatic Differential Equation Solver (C. Dill, C. Ellis, K. Ratliff)

The following is a summary of changes which have been made recently:

1. The COMMON area in the program has been rearranged due to problems of adequate space for the COMMON variable.

2. The User Manual for the program has been revised and added to the card deck of the program for user convenience.

3. The provision in the Supervisor for specification of format for output (page headings and number of lines per page) was not functioning correctly since a standard heading was being used in all cases. This error has been corrected.

4. Another error was detected in the evaluation of partial derivatives when the differential equation is given using single letter variables for simplification of the equation. Since the values of the auxiliary variables are computed by the function evaluation subroutine (SUBFUN), these values when used in the partial derivatives, were often outdated. This is because they were evaluated using the values of the independent and dependent variables at the last call to SUBFUN. This has been corrected by calling SUBFUN immediately prior to the evaluation of the partial derivative matrix. This will not introduce much additional computing time since the matrix is seldom evaluated.

5. A bug has been found and corrected which was causing inaccurate evaluation of both the function and its partial derivatives. It was related to the number of registers used for computing these two things, and only occurred in the case of very complicated differential equations.

A few small changes have also been made in the method of integration. A master copy of the program has been made and is currently in the process of being transferred to a tape via the ASP card to tape program. This will facilitate making tape copies of the program in the future using the ASP tape to tape program.

3.1.3 Application of the Numerical Method (C. Gear, K. Ratliff)

A set of 26 equations describing the post-detonation reactions of the elements in the upper atmosphere was investigated this month. Previously, they had been taking 30 minutes of CDC 6600 CPU time. Our program would handle them in about 1.5 minutes of 360/75 time and there have been indications that its run time on the 6600 is negligible compared with the compile time. The complexity of the equations underlined the problem of determing corrector convergence, and it was necessary to modify the criterion and to require more frequent re-evaluation of the partial derivatives. They are now recalculated whenever the step or order changes as well as when convergence fails. (Calahan of the University of Michigan has found it advantageous to re-evaluate at every correction step because he is using a sparse matrix inverter which is fast for the type of equations arising in network analysis.)

3.2 Graphics

Two distinct thrusts were underway this quarter: the extension of the DEC 338 program to handle general purpose diagram constructions (it will be used as the graphical front end to the proposed simulation program--see File No. 785), and the preparation of programs for the 360 to replace those that used to exist on the Illiac II.

3.2.1 DEC 338 Software (T. Y. Lo)

The work on the implementation of the general purpose display package is proceeding with the coding and debugging of the routines to provide the facilities for a user to construct his own symbols. The symbols constructed in this step will be used as light-buttons to construct a model of his design problem in the next step. In addition to the features mentioned in the last quarterly report, a new feature is to be implemented into the package so that a user is able to use other light-buttons to form a part of a new light-button under construction. This feature of using light-buttons to be parts of a light-button is planned to be available also in the second step in using this display package, i.e. when the user is constructing a model of his design problem, he can replace a group of light-buttons (not necessarily different) and the connections among them with a new light-button to be defined by the user at this step. This feature will be very sueful when a user is constructing a rather complicated design model using some simple light-buttons that he has defined in the first step.

For example, when a user is constructing an electronic network, he may define a transistor symbol as a new light-button to represent a group of symbols so constructed as to be an equivalent circuit of the transistor. Then in constructing the network, the transistor symbol can be used to replace its equivalent circuit. In order to be able to handle these light-buttons properly, a lot of tables are constructed during both steps in using this display program. Since most of these tables are of variable length, it is worthwhile to compact them after the construction of light-buttons is done. At the time the user requires modification of his light-buttons, the tables will be scattered again so that necessary changes can be made on the tables easily.

A detailed description of this piece of program will be provided after the debugging and testing of the routines are completed.

CAPS 338 Software

The "ADD" operation of the flowchart editing system of CAPS has been implemented during this quarter in order to complete all the operations as described in appendix B of the Department of Computer Science Report No 257.

3.2.2 FPL-II (S. Wilkins)

The major part of the conversion and modification of the flowchart programming system has been completed. The largest incomplete section is decision processing.

Master Error Control Program

During the execution of a flowchart any number of unusual conditions may occur. These conditions must be recognized and corrected if possible, and the user must be informed of the actions taken by the system. The following table describes the errors that can occur and the action taken. The control of these exceptional conditions is now operational. Control of other exceptions will be added to the system as the need arises.

CODE	TYPE	MEANING	ACTION
8	Fixed-point overflow	Overflow occurred during fixed-point add or subtract	The format of the data is changed to floating-point and the operation is repeated as a floating-point operation. The user is not informed.
9	Fixed-point Divide	Quotient of a fixed-point divide cannot be expressed as a 32 bit integer	The format of the data is changed to floating-point and the division is repeated in floating-point mode. The user is not informed.
10	Illegal Operation	For the operations: ADD, SUB, MPY, DVD, SET, PRINT one of the following occurred 1) Operand not allowable type 2) Invalid mixed operands 3) Invalid operator	Error is unrecoverable. Current user is aborted. Provides message: "ILLEGAL ØPERATIØN SYMBØL NØ X* ØPERATIØN Y** JØB TERMINATED"
11	Division by Zero	Attempted division by zero operand in fixed or floating-point mode	Error is unrecoverable. Current user is aborted. Provides message: "DIVISIØN BY ZERØ SYMBØL NØ X JØB TERMINATED"
12	Exponent Overflow	Characteristic overflows on floating-point ADD, SUB, MPY, DVD	Error is unrecoverable. Current user aborted. Provides message: "EXPØNENT ØVERFLØW SYMBØL NØ X ØPERATIØN Y JØB TERMINATED"
13	Exponent Underflow	Characteristic underflow in floating-point MPY or DVD	Error is recoverable. Result set to zero. Provides message: "EXPØNENT UNDERFLØW SYMBØL NØ X ØPERATIØN Y RESULT SET TØ ZERØ"
14	Significance	Intermediate result of addition or subtraction is zero in floating-point mode	Error is recoverable. Result is set to zero. Provides message: "SIGNIFICANCE LØST SYMBØL NØ X ØPERATIØN Y RESULT SET TØ ZERØ"

* X symbol number of flowchart symbol currently being executed

** Y can have values +, -, *, /, SET, PRINT when applicable

The next stage in the development of FPL-II is the implementation of decision processing.

Decision Processing

The decision symbol, the diamond-shaped processing box, allows the user to specify alternate paths to be taken in his flowchart at execution time depending on the results of previous processing.

Two primitives are provided to be used with data to be tested in the text associated with the decision symbol.

In the post-fix notation currently implemented the primitives are:

1. SIGN

   Used to test the sign of an arithmetic expression

   ex. A SIGN (tests sign of data with name A)

   AB + 2/ SIGN (tests sign of expression (A+B)/2)

   Returns values corresponding to +, -, 0.

2. COMP

   Used to compare the values of 2 variables or a variable and a constant.

   Allowable data types for comparison are:

   a) FIX & FIX
   b) FIX & FLOAT
   c) FLOAT & FLOAT
   d) FIX & numerical constant
   e) FLOAT & numerical constant
   f) CHARACTER STRING & CHARACTER STRING
   g) CHARACTER STRING LITERAL (enclosed in single quotes) & CHARACTER STRING

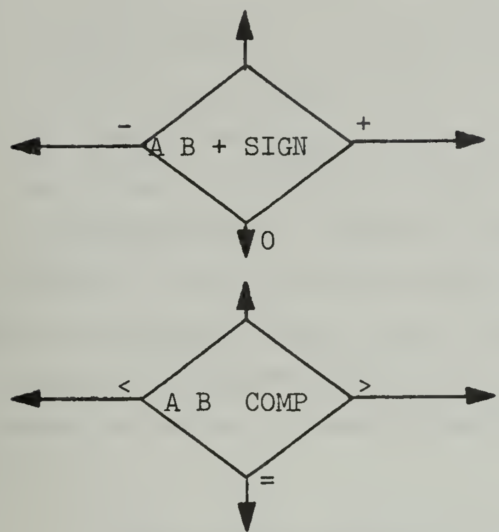
   ex. A B COMP (compares 2 variables of allowable types)

   A 2 COMP (compares numerical variable with constant 2)

   A 'BCD' COMP (compares character string variable A with character string literal 'BCD')

   Returns values corresponding to > (1st greater than 2nd)
   < (1st less than 2nd)
   =

The primitives are used in the following examples:



The paths to be taken as a result of the test are indicated by associating the symbol for the result with the appropriate arrow from the decision symbol.

(Not all the test values for a primitive need be used at a decision symbol.)

If one of the arrows from the decision symbol is left blank and the test fails for marked arrows then the path indicated by the blank arrow is taken.

If there are no blank arrows and the test fails for the marked arrows then processing stops.

3.2.3 <u>360 Graphics Monitor</u> (M. Michel)

The following summary of the current state of the 360/50-PDP8 interactive graphics monitor system consists of two parts. The first is a brief discussion of the general characteristics of communication between the PDP8, the PDP7, and the 360/50. Its purpose is to acquaint the reader with the mechanics of the communication problem involved and to provide him with a feeling for the situation. The second section describes the structure, functions, and utility of the monitor system as currently implemented and envisioned, and states the current status of the project.

3.2.3.1 <u>Background Information for 360/50-PDP7-PDP8 Communication</u>

A. PDP8 to PDP7 to 360/50

The PDP8 transmits or receives one character at a time to or from the PDP7 at a maximum rate of 100 characters per second. As soon as the PDP7 senses a carriage-return character from the PDP8 or when it has

received a total of 125 characters without getting a carriage-return character, the PDP7 considers the "line" of characters from the PDP8 to be completed. At this time, the PDP7 transmits a carriage-return followed by a line feed to the PDP8. The PDP7 then tries to send the line to the 360/50. When the 360/50 reads the line, and not before, the PDP7 sends a bell-character to the PDP8, signifying that the PDP8 may send another line. If the PDP8 should try to send characters to the PDP7 before receiving the bell-character, the PDP7 immediately sends a carriage-return line feed, '#WAIT', carriage-return, line feed sequence to the PDP8.

The format of the line sent to the 360/50 from the PDP7 is as follows:

```
A,B[125 data bytes]CR
0 1 2            126 127
```

Where A and B (bytes 0 and 1) are PDP7-360/50 control characters, byte 127 is always a carriage-return, and bytes 2 through 126 are data bytes. If the PDP8 had sent 50 characters followed by a carriage-return, the line would look like:

```
A,B[......CR---Junk---]CR
0 1 2    51 52      126 127
```

B. 360/50 to PDP7 to PDP8

When the 360/50 sends data for the PDP8 to the PDP7, the data format is:

```
AB[N---124 data bytes]CR
01 23              126 127
```

Where A and B are PDP7-360/50 control characters, N is a control character for the PDP8 monitor system, and byte 127 is always a carriage-return. If 50 characters were to be sent to the PDP8, the line would be:

```
AB[N    CR---Junk---]CR
01 23 52 5354      126 127
```

The PDP7 would send only bytes 2 through 53 to the PDP8, followed by an additional line feed [Note: when the PDP8 initiated a send, a line feed

and a bell were returned; when the /50 initiates a send, only a line feed is additionally sent to the PDP8].

After sending one line, the /50 cannot send another line until the first has been completely sent (character by character) to the PDP8. Note that the PDP7 assumes the PDP8 is a synchronous device; hence, the PDP7 sends characters to the PDP8 at a fixed rate and the PDP8 must accept them at that rate. In order to know when this has been accomplished, the program in the /50 must issue a READ operation. In this case, byte 1 ('B') of the input indicates the disposition of the last line. (On ordinary input, B = 0, meaning the line is a data line). The rest of the line is junk. If B is

1, then the output has all been transmitted to the PDP8.

or 2, then the PDP8 has requested the PDP7 to stop transmission before the entire line was sent, i.e. the remainder of the line has been voluntarily lost;

or 3, then the PDP8 has "signed off."

Hence, to send several lines to the PDP8, the program in the /50 must issue a read before every output operation (except the first) and check B for the proper setting.

C. Data Format

Since certain characters, such as carriage-return and linefeed cause the PDP7 to take some action, data--particularly object code--cannot be sent 8 bits at a time from the PDP8; nor is 8 bits really convenient considering the 12 bit word of the PDP8. In addition, the 360/50 cannot send data in its original form to the PDP7 either, since an inadvertant carriage return in the middle of the line would cause the PDP7 to stop transmitting to the PDP8. Hence, all data (except pure character data sent only to the time sharing system) is sent in a coded format, six data bits (half of a PDP8 word) per 8 bit character at a time. Since internal PDP8 characters are six bits and the transmitted form takes up 8 bits (1 byte), conversion in the 360/50 to EBCDIC is very neat and clean. Numeric data is reformed into 360/50 half words, i.e. one PDP8 word of 12 bits becomes 2 bytes. In the PDP8, each incoming data byte is stripped of the excess 2 bits and every 2 characters are packed into one PDP8 word.

3.2.3.2 Graphic Monitor System Overview and Status

A. Overview

The PDP8-360/50 monitor systems and 360/50 I/O macros have been designed to help reduce the bother and complication inherent in the 360/50-PDP7-PDP8 communication linkage. The package consists of communication and debugging structures embedded in a double ended monitor system, one end in a non-terminating partition in the 360/50, the other end permanently resident in bank 3 to the PDP8.

At the highest level, a user sitting at the PDP8 teletype uses the PDP8 monitor to communicate with the 360/50 monitor. On command, the PDP8 monitor enters one of several modes: it can become a simple console plugged into the time sharing system, or it can command the 360/50 to load a user module into the 360/50 graphics partition or to load data into PDP8 core, etc. The following types of operations are done at this level.

1. Using the time sharing mode, the user can create symbolic source and data files, enter jobs into the regular ASP job stream, etc. Output from these operations is transmitted back to the PDP8 and displayed on the 338 scope for easy viewing. Concurrently, the output can be routed to dectape, to disk (PDP8 or 2314) or 1403 printer for hardcopy version.

2. Using monitor-to-monitor communication, the user can call non-interactive, service programs into the graphics partition. As an example, PDP8ASM (a 360 program for assembly of PDP8 source code) could be called in to assemble a PDP8 program from a source file created by use of the time sharing facilities. The object code is stored on 2314 disks, the listing is sent to 2314 disk for "off line" printing and/or to 338 scope for immediate viewing (instant turn-around). The user notes his assembly errors, calls time sharing, changes some statements in the source file, recalls PDP8ASM to try again (the first object file and source listing can be overwritten, thus eliminating unneeded printout, et al). Depending on partition size the user could call in <u>any</u> program, enabling him to have instant turn-around, for example, for any language on the system, or for any analysis program, etc. The user can then command the 360/50 monitor to pass files to the PDP8 (source, object code, data, et al). These can be added to the PDP8 disk file or tape file (using dectape).

3. Interactive programs can also be called into the graphics partition (this was, naturally, the main intent all along). The usual sequence would be something like: assemble and perfect programs for the 360/50 and for the PDP8 in 1 and 2 above. Load the PDP8 programs into the PDP8 and add to dectape. Call in the new module just created into the 360/50; call in the new PDP8 programs. We now have user application programs running at both ends of the system.

Quite a different level of operation is now in effect; two application programs are communicating with each other and with the user. The 360/50 programs use the same I/O macros used by the 360/50 monitor. The PDP8 programs use the I/O routines of the PDP8 monitor, namely 'loader' and 'sender', for communication with the 360/50 and a system message 'SYSMSG' routine for communication with the user. The PDP8 monitor maintains control of the interrupt structure, sending control to user and system routines as appropriate. In particular, strict control is maintained over PDP8-360/50 communication; namely, PDP7 messages and 360/50 system messages are routed to a special buffer for display to the user. When these occur, certain functions such as transmission from the PDP8 to the PDP7 are inhibited until the user takes corrective action. This may include waiting, retrying the operation, cancelling the operation, specifying new parameters, adding parameters, etc. All PDP8 system routines are easily accessible to user routines via calling sequences or parameter settings. The user can get back to the monitor system at any time, either from his program or by a manual interrupt.

If a user bomb-out occurs in the 360/50, control is automatically passed to a resident debugging routine; this routine communicates with a special section of the PDP8 monitor and is completely transparent to a user program in the PDP8. In brief, the package allows the user to "scroll" around through 360/50 core, to pointer chase, to take "snaps," to patch 360/50 core, and to restart his 360/50 program, all from the PDP8/338.

B. Status

As of this date both monitor systems and the 360/50 I/O macros are written and are in "middle" to final debugging stages, i.e.' they are usable but not for production operation. PDP8ASM converted from an assembler for PDP7 programs is usable but needs much polishing and several additional features for maximum utility. The debugging package has been specified but not yet implemented. One necessary component of the package is a text editor, which, hopefully, can be partly "stolen" from existing programs for the PDP8. An important aspect of this package is that it will be written as a user program for the system and will be carefully documented; hence, it will serve as a model demonstrating what can and has to be done in order to "hook into" the system. Several useful utilities for manipulating input/output between the PDP8, 360/50, printers, disk storage, etc., which would be made part of the monitors or callable by them in addition to the basic facilities of the system, have been considered but have not yet been implemented due to lack of time. Another very useful addition would be a 360 FORTRAN interface to the I/O so that FORTRAN programmers could utilize FORTRAN type I/O readily. Finally, a user program running under the PDP8 monitor must be written to interface the system with the flowchart interpreter system.*

* Suggestions for additional features are welcome.

3.2.4 DRAWL-Graphics Display Language (J. Roman)

Batch-processing GDL was made fully operational and will shortly be made available to the general 360 user as a catalogued procedure. A manual describing the use of GDL has been completed and will be made available about the same time.

Work on an interactive version of GDL has been started and will use the PDP-8 and 338 equipment. Because of its large size GDL will reside in the 360/50 while input will come from PDP-8 via PDP-7 and output will be available at the teletype and 338 display unit. GDL has been modified to function in this environment and I/O routines to handle data transfer have been written and are now in the final testing stage. Since free-format input seemed necessary in an interactive environment, a translator routine has been written (also in the final testing stage) permitting free-format input at the teletype as well as some error-checking capabilities.

3.3 PLORTS Time Sharing

A test version of PLORTS Phase 0 (filing system and remote job entry) is now on the air. Several major holes in the system must be plugged before it is released to the users. A number of further modifications to ASP have been made. Pass I of the compiler/executor is now nearly complete. Work has continued on Pass II and Pass III, and the definition of the interface between them has been revised. The first version of the filing system is in use.

3.3.1 ASP (P. Boekhoff)

Two DSP's, PLORTS and FILE, have been added to the working ASP 2.0 system. These perform input from the PLORTS filing system to ASP and output to the filing system, respectively.

The PLORTS DSP is logically parallel to the stock RDR (card reader) DSP. It receives its input from the PLORTS filing system in Partition 1 of the 360/50 as a sequence of card images. When a terminal user (and there are several) enters the command

RUN JCLFILE,SOURCEX,JCLEND (zB)

the PLORTS executor in Partition 1 (ultimately Pass III; this is presently being simulated by a limited executor which handles filing and RJE commands) will read the named files and pass the concatentation thereof to ASP in Partition 0. The PLORTS DSP then generates a job number in a format compatible with that assigned by the department for jobs submitted at the routing room, returns this number to Partition 1 for transmission to the terminal as a job receipt, and "reads" the job from Partition 1 into the ASP input stream.

The FILE DSP is invoked via a /*PROCESS FILE statement and appropriate /*FORMAT statements in the input stream. It may be called by a job submitted at the routing room or by one entered at a terminal. Allowable parameters on the /*FORMAT FILE card are:

DDNAME=	dd name of dataset to be filed
DEST=	name under which the dataset is to be filed. FILE takes the PS# and user name from the job's /*ID card and adds (concatenates) the DEST= data thereunto. The DEST= parameter may be up to 32 characters.
CONTROL=SINGLE/DOUBLE/PROGRAM	If this is omitted or PROGRAM is specified, the first character of each record is assumed to be a control (carriage or stacker select) character and is stripped off before the record is passed to Partition 1. Otherwise, the record is sent unchanged.

Any other parameters will be politely ignored.

The FILE DSP sends an ID record to P1 at the beginning of each dataset. If for example the /*ID card specified 153 and FILLMORE for PS# and user name, and the /*FORMAT card specified DEST=ST.YONKERS, the ID record would be

PS=0153ID=FILLMOREℓST.YONKERS

where ℓ is a byte giving the length of the succeeding DEST data (in this case, 10). This dataset would then be filed as 0153.FILLMORE.ST.YONKERS by the filing system.

Most of this quarter's messing with ASP consisted of changes to Input Service to recognize these parameters for /*FORMAT FILE.

3.3.2 PASS I (J. Christopher)

The storage CSECTs and DSECTs have been modified. All commonly used V-type literals now appear in the PASSI DSECT, thus conserving storage. The line table and symbol table have been moved into the output storage section. The portion of the PASSI DSECT which is common to Pass II now appears first.

The following keyword-statement syntax check routines are presently debugged: ALLOCATE,IF,DO,CLOSE,RETURN,CALL,GO TO,OPEN,END, BEGIN,EXIT,STOP,KEYWORD, which recognizes statement-type keywords, and LHSIDE, which analyzes the lefthand side of assignment statements and all labels and determines whether a statement is a keyword statement or an assignment statement.

A minor bug on the outputting of operators in suffix notation has been corrected. The LHSIDE routine has been updated so that it checks for condition prefixes and ignores them. It determines the statement type in the following manner. If any identifier is seen, subscripted or not, the pointer to the first character of the identifier in the input buffer is stored, the identifier and subscript (if present) are analyzed as usual, and the following character is tested. If it is a colon, a label is assumed; if an equal sign, the lefthand side of an assignment statement is assumed; a comma is assumed to indicate multiple assignment, which is a fatal error. Otherwise a keyword type statement is assumed, and control passed to KEYWORD with the current character register containing the first character of the identifier. The input string pointer is backed up to the stored value. If a possible keyword is seen, i.e. an identifier consisting of more than six alphabetic characters, the character following it or following a proper nest of parentheses following the identifier is tested (a proper nest of parentheses is a string of characters, the first of which is a left parenthesis, the last a right parenthesis, and an equal number of left and right parentheses inbetween). If it is a comma, the action is as above. If it is a colon or equal sign, the possible keyword is assumed to be an identifier of more than six characters, and

is truncated to six and accepted and a message to that effect is produced. This is done by rescanning the identifier with the "identifier only" flag set. Ohterwise, a keyword is assumed and since the identifier is already in the identifier buffer, control passes to the secondary entry point of KEYWORD which does not call IDENTFYR.

A structure similar to KEYWORD--an entry point which does not call IDENTFYR--has been added to ATTR, the attribute recognizer.

LINEND1 now interacts with GETCHR to produce the following results for nonblank characters after the first semicolon. Comments are scanned and ignored; if any other characters appear a warning is produced to the effect that input after the first semicolon on a line is ignored.

As a consequence of the above proviso, if an ELSE clause is given with an IF statement, it must occur on the next line of the program. The method for handling and outputting IF statements has been revised. The output format is

{statement type code} {label information} {expression}

{statement}, where {statement} is the statement following THEN.

After THEN or ELSE is seen a flag is set and a phony entry in the line table is constructed, and the rest of the statement is handled as a complete statement. If it is a keyword statement the type code is checked in the KEYWORD routine. If it is any of IF,DECLARE,END,PROCEDURE,ENTRY or ELSE, a fatal error is flagged, since these statements are presently not allowed to follow THEN or ELSE. Otherwise, the statement is analyzed and LINEND1 checks the THEN-ELSE flag and if set removes the phony line table entry.

The format for the output of DO statements has been revised. It is now the following. If the DO statement is DØ, the output is

{statement type code} {label information} X'00' {End-of-Statement}

the code byte is zero. If the DO statement is DØ WHILE (expression), the code byte is set equal to two, and is followed by

{expression} {End-of-Statement}.

Otherwise, a type of code of one is used for each specification. In this

case the format is

{identifier} {subscript (if any)}{expression} {pointer to secondary string location for TØ expression, if it occurs} {pointer to WHILE expression, if it occurs} {BY expression}

If TO occurs and BY does not in a specification, a default condition BY 1 is assumed. If either TO or WHILE is omitted the corresponding pointer byte is set to zero.

The OPEN statement format has also been revised. Each list produces a 3-byte attribute table as in a DECLARE statement, followed by the internal form of the file name. Unimplemented options are recognized and ignored. After a given keyword in an OPEN statement list is checked for unimplemented options, control is transferred to the secondary entry point of ATTR, and the attribute is handled as if it occurred in a DECLARE statement. At the end of the list, the attribute table is checked. If any attributes not legal in an OPEN statement occurred, an error message is generated to that effect.

The PROCEDURE statement routine is almost debugged. Its format has also been revised. After the label information occurs a byte which contains the OPTIONS(MAIN) and RECURSIVE bits. Then occurs a 3-byte attribute table, followed by the internal form of parameters, if any. After the parameter list has been output, each successive keyword is checked. If it is OPTIONS(MAIN) or RECURSIVE the appropriate bit is set. Otherwise, the attribute is handled as if it occurred in a DECLARE statement. At the end of the statement the attribute table is checked. If any attribute other than arithmetic or bit string is found, a fatal error is generated.

A routine to eliminate wasted space in the output buffer has been written and debugged for special cases, but reorganization of the buffers will be necessary for it to work generally.

Line numbers are now handled properly. Pass I is complete except for the following:

1. No special action is taken for unnumbered lines. Certain unnumbered statements will require special treatment for proper functioning of the interface between the compiler and the timesharing system.

2. Constants must be generated and output into the symbol table, and the symbol table pointers must be inserted into the Pass I output buffers.

3.3.3 PASS II (J. Christopher, A. Whaley)

The symbol table entry format has been revised. It is now 12 bytes in length.

BYTE 0

bits 0, 1, 2	0	constant
	1	variable
	2	statement label constant
	3	dummy parameter
	4	operator
	5	entry name
	6	procedure name
	7	pseudo-invocation
3, 4	0	character string
	1	bit string
	2	binary fixed
	3	bianry float
5	0	real
	1	complex
6	0	nondimensioned
	1	dimensioned
7	0	not assigned
	1	assigned

BYTE 1 (Character string or bit string) Length minus 1

BYTE 2

bit 0 (if = 1) parameter		
bit 1 (if = 1) *subscript		
bit 2 (if = 1) label variable		
3, 4	0	input file
	2	output file
	3	print file
5, 6	0	automatic
	1	static
	2	controlled
7	0	single precision
	1	double precision

BYTE 3 unused

BYTES 4→5 pointer to location in core (not set for variables in Pass II)

BYTES 6→11 EBCDIC characters of identifier padded on the right with blanks, except for constants (in which case these bytes contain the value and as many bytes as needed are used) and dummy variables in which case these bytes contain EBCDIC blanks.

A "pseudo-invocation" is merely an occurrence of an identifier which has not been explicitly declared or has apprered as an entry name used in a CALL statement.

The Pass II expression analyzer MOVEX has been written. It transfers the expression from either Pass I output buffer into the Pass II output buffer. The address of the start of the input buffer is input in R3 as a parameter. The routine does the following. It inverts all identifiers into symbol table (SYMTAB) pointers. When an identifier is seen, it is converted to EBCDIC in a standard buffer (TRANBUF) and the input pointer (R3) is advanced beyond it. Then the routine SST is called, which searches the entire SYMTAB for the given identifier. If found the symbol table pointer is output into POLIII, the Pass II output buffer. If not found, the routine EXTEN is called which makes a pseudo-invocation symbol table entry for the identifier. If a symbol table pointer is given in the input string, the two leftmost bits are turned off and the pointer is moved into POLIII. The above is done until an End-of-Statement marker or an All-Zero byte not part of an identifier or SYMTAB pointer is reached.

All the table searching routines have been written. SST and SCB have been described previously.

A routine has been coded to handle all statement labels. In the case of a subscripted label, a pointer to the beginning of the statement in the Pass II output string is placed in the storage cell allocated for the given element of the array.

ENTRY names are handled similarly to label constants. Whenever an explicit declaration of an entry name is seen, and a symbol table search finds a pseudo-invocation entry for that name, the symbol table entry is converted to a PROCEDURE or ENTRY, as appropriate.

The Pass II PROCEDURE statement has been modified. It now begins by generating a GO TO which points to a generated statement-label-constant symbol table entry. A pointer to this entry is also placed in the ENTAB stack delimiter for the procedure. When the END statement for the procedure is encountered, ENDIIST fills the value of the label constant. This allows control to pass around the procedure. An ENTAB stack delimiter is now output for each ENTRY name in the PROCEDURE statement. All but the first have flag bits set, to indicate how many must be removed.

It has been necessary to add level numbers to PASSII for statement label constants, ENTRY names and PROCEDURE names. The level is kept in storage. Each time a new PROCEDURE or BEGIN block is encountered, this number is incremented by one and stored in the symbol table entry for all entry names of the block and label constants declared internal to the block. This level number serves two purposes. Upon reaching a statement which terminates a block during execution, all automatic storage for that block must be released, i.e. storage with the highest level number. However, when a statement is reached which terminates several blocks (a GO TO statement or END <procedure name>), the execution phase must release storage for all automatic variables with level numbers greater than that of the label in the GO TO or the procedure named in the END statement. Since all references are resolved in Pass II, the level numbers must be generated in Pass II.

A method has been devised for finding unresolved forward references in GO TO statements. It involves a routine called SBNEST, which expects two parameters--an identifier in TRANBUF, and a block table pointe in R13. It searches the block indicated by the pointer and all nested blocks for an occurrence of the identifier in TRANBUF. It has pass and fail exits.

The Pass II GO TO statement routine GOIIST has been written. For GO TO <identifier>, it translates <identifier>, and then uses SST to determine if the <identifier> has already been declared or referenced in a containing block. If so, a pointer to that symbol table entry is output. If SST does not find the identifier, however, the symbol table sections for

all block nests contained in the currently open block at that point in Pass I are searched for the given identifier. If it is a statement-label constant entry with no value, then a pointer to this is output since the entry was generated by a forward GO TO. Otherwise, a symbol table entry is generated for a statement-label constant with no value.

The subroutine FWBWI is employed which returns the block table pointer to the currently open block.

The action of GOIIST is simpler for a reference to a subscripted label variable. SST is called to find the identifier. If not found, a fatal error is flagged. If found, its type is checked. If it is not label array, an error results. Ohterwise, the pointer is output. Statement-label variable references are handled as statement-label constant references.

The Pass II END statement pops the entry name stack and removes stack delimiters. A method for deciding which blocks it should close has also been devised. If END <entry-name> is seen, the symbol table is searched for the entry-name. If not found, an error condition is raised and compilation terminated. If found, its core pointer is compared with those for all blocks, and the "closed" bit is set for all blocks whose beginning indices are greater than or equal to that of the entry name.

The Pass II DECLARE statement routine has been flowcharted.

3.3.4 PASS III (A. Whaley)

Data conversion routines have been written to handle data type conversion for the entire PL1 timesharing system. These routines accomplish type conversion of all types necessary for processing assignment statements, for constructing print lines for input/output operations, and for performing number conversions on constants appearing on source program statements.

For ease of programming and to reduce program size, the four basic data types are handled as follows:

1. Binary Fixed Point. Only one data length, or "precision", is allowed for fixed point numbers. They are stored as 32 bit integers in two's complement form. No fractional precision may be associated with a fixed point number, as is allowed in the IBM version.

2. Binary Floating Point. Floating point data may be 32 or 64 bits, corresponding to standard 360 single and double precision.

3. Character Strings. Length of the string may be from 0 to 255 characters. The PL1 VARYING attribute may be used with character strings. Character strings are stored as EBCDIC characters.

4. Bit Strings. This is the only type handled in a non-standard fashion. Rather than storing a bit pattern, 8 bits per 360 byte, bit strings are kept in the original form in which they appeared on the source statement. This results in a character string consisting of the character representations of ones and zeros. For this reason, all string handling logic is similar, requiring only a check for valid characters in the case of bit strings. Valid lengths are from 0 to 255 bits (EBCDIC ones and zeros).

No other data types are handled in this subset of PL1. The conversion of these four types is handled by sixteen separate conversion routines. The appropriate one is selected on the basis of the type of input data and the desired type of the output data. Both length and type conversion are handled simultaneously.

Several of the sixteen routines are called individually by Pass I of the compiler to facilitate conversion of constants input on source statements. The format item operators and DATA- and LIST-directed transmission routines also invoke these sixteen routines to produce or decode character representations of the data types they handle.

During processing of assignment statements, only coded descriptors of the data involved are immediately available. Therefore, a central routine known as TYPCØNV is called with two of these descriptors. The routine analyzes the descriptors, and constructs actual storage addresses and lengths and invokes the correct routine of the sixteen available to handle actual conversion. Allocation of storage is also handled if the destination descriptor indicates that this is necessary.

TYPCØNV also handles the conversion of complex data by forcing it to appear as two separate items of non-complex form.

3.3.5 Phase 0--RJE and Filing System (A. Whaley, N. Weidenhofer)

The test version of the Remote Job Entry system which uses the BLAM (Basic Line-numbered Access Method) filing system is now on the air on a permanent basis and functioning well enough to allow a user to build a file, modify it, and submit it to be run in the batch system. On the negative side, crashes are still frequent and file integrity is below the acceptable level. Programs to dump the contents of the disk on tape and reload it have been written and work most of the time. Since the dump and load programs have been written to deal with the saved files rather than with disk blocks (logically rather than physically oriented), they also serve to clean-up the files to some extent and compact them so that they occupy less disk space. As soon as they are more completely debugged their regular use should go far toward improving file integrity

To log on to the timesharing system, the user should do the following:

1. Turn on the teletype and press the carriage-return key. The system should respond:

```
#<console number> TIME SHARING ON
ENTER PS#, USER NAME
```

2. The user must then type in his problem spec number (leading zeros not required) and name by which he is known to the system. The user name is limited to eight alphanumeric characters--periods, blanks, and other special characters are not allowed and may not be "protected" by apostophes.

The RJE system accepts the following commands:

RUN <file name>, <file name>,...

"RUN" concatenates the files named and transmits them, line-at-a-time to ASP to be submitted as a batch job. Each line of the several files is truncated or padded to eighty characters with the last eight characters being the line number.

DEST <file name>

"DEST" (DESTRØY) causes the file named to be deleted from the user's catalog and the space used for it to be released for re-use.

LØGØUT

LØGØUT closes the user's catalog and logs the user out from the system.

ØPEN <file name>

"ØPEN" issues an open command to the file system so that the user can begin to build and/or modify the file named. If a file with that name does not already exist in the user's catalog, one is created.

CLØSE

"CLØSE" closes the file the user currently has open, allowing him to issue another "ØPEN" or a "RUN" command.

LIST <line#> <line#>

"LIST" causes those lines in the file whose line numbers fall between the numbers specified to be copied to the user's terminal. Both line numbers are optional; if one is supplied, the file is listed from that line to the end. If neither is supplied, the entire file is listed.

DEL <line#> <line#>

"DEL" (DELETE) causes those lines whose numbers fall between the numbers specified to be removed from the file.

A line may be added to or changed in a file by typing the line preceded by the line number and a space.

Besides handling user's terminals, the RJE system accepts data from ASP, putting it into a user's file as described earlier. Each record in the data set transmitted by the FILE DSP is preceded by a line number generated by the file system and written into the user's file. The line numbers begin at 1.000 and are incremented by 1.000 for each record.

Among the holes to be filled are:

1. listing the user's catalog.

2. improving file integrity, especially around crashes

and several others

3.4 Computer Design

At a theoretical level, techniques are being explored for the "optimization" of microprogram control parameters, while at a practical level, a small remote graphics console is being designed.

3.4.1 Microprogram Optimization (L. Greninger)

Of the three ways mentioned in the last report for minimizing the number of bits of control memory necessary to hold a given micro-programming (minimize the number of instructions, minimize the width of the instructions, and maximize the multiple use of instructions), work this quarter concentrated in the first, minimizing the number of instructions. Since a formal automatic minimization of this sort takes too much time, work has been directed toward an interactive system where the messy and time consuming cases are handled with the aid of the user.

The system envisioned will use the PDP-8/DEC 338 as the man-machine interface but will run on the 360/75. The program to be minimized will be stored as a Sutherland flow graph. The user will draw on the CRT a flow graph F together with a second flow graph F'. The system will then locate all instances of F in the stored program and replace them each by an instance of F'. It will be possible to input a number of these flow graph pairs along with directions as to the order and circumstances under which they are to be applied to the stored program.

Once the system has done the requested substitutions it will display on the CRT the effects of each of the substitutions it has done. If the user sees an instance where some different substitution might have been done with better results, then he can stop the system, input a new

pair of flow graphs with new directions, and ask the system to try again.

When the user is finally satisfied that the program is minimal, the system will transform the flow graph of the program back into a sequence of instructions utilizing a minimum number of registers. It will also attempt to reformat the instructions so as to minimize their width.

The heart of this system is a graph storage and manipulation system. This system, called GRAPH4, is designed to handle the building, storing, and modification of graphs, subgraphs, and graphs of graphs, as well as subgraph recognition and extraction through template matching.

GRAPH4 is currently in its fourth iteration. It is being programmed in FORTRAN IV using WATFOR due to the excellent debugging facilities available. An attempt has been made to map the FORTRAN routines into macro calls, but the 360 macro format has proved too restrictive to make the resulting gain in efficiency worth the effort. Dissatisfaction with the 360 macro system lead to a survey of all literature available on macro systems. Many fine systems were found but none of them were done on the 360. Time limitations precluded writing a new macro system, so programming for GRAPH4 continues in FORTRAN IV.

3.4.2 Graphical Input Device to the Display Terminal (J. Rettig)

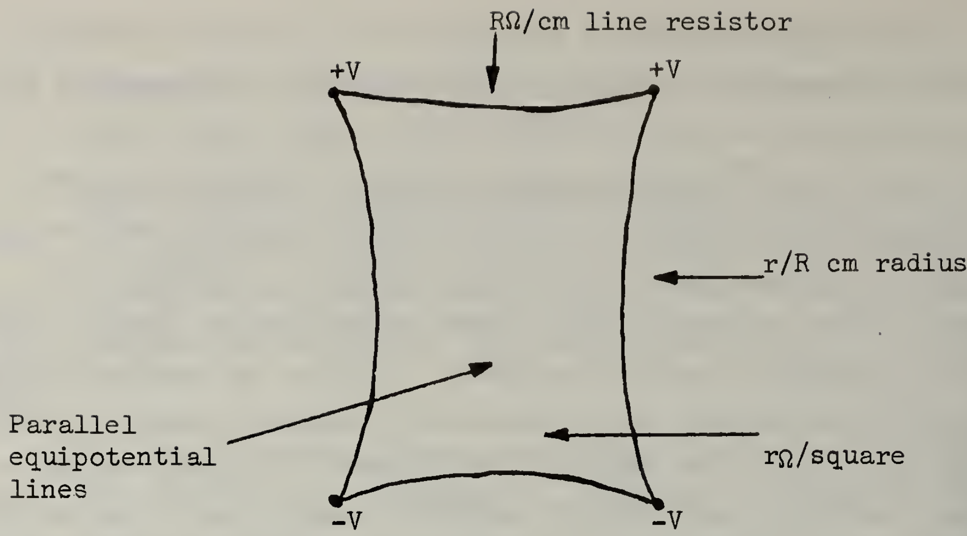
The A-D convertor which will be used to track our analog input device has been designed and built. The x and y analog voltages are converted during each sector gap time and the tracking cross is displayed during one of the sector gap times. The x and y display registers are used in the conversion. Therefore, the contents of these registers must be placed in temporary registers during the sector gap. During the sector times, the x and y analog input voltages will be compared with the x and y voltages applied to the display, and an interrupt will be generated when they are equal. During each sector gap a signal will be generated by the convertor to indicate that the A-D conversion is complete. At this time the CPU can obtain the position of the analog input device.

An incremented joystick has been built and will be tested as soon as the A-D convertor is working satisfactorily. The possible use of a resistance sheet as input to the display scope is also being considered. A 4" by 6" piece of stannic oxide coated glass has been tested. The resistance (80 Ω/sq.) of this sample has been found to be uniform and the coating seems to be fairly durable. Currently, no satisfactory method has been found to make contact to the stannic oxide coating with negligible contact resistance. We plan on spraying contact pads of silver filled epozy on the glass and clamping a piece of copper foil on top of the pad. We must also decide upon a suitable stylus to pick up the x and y voltage from the tablet. Three types of stylus are being considered: a lead pencil, a ballpoint pen, and a graphics pen.

Other types of resistive sheet material such as plastic with graphite embedded in it and glass with a chromium coating are also being considered. Glass has the advantage over the plastic sample of transparency. Thus, images can be projected onto it and traced. However, a scratch on the glass may cause a discontinuity in the current flow and in the x and y coordinate voltages. We have obtained a sample of the chromium coated glass and we plan to test the uniformity and durability of the surface.

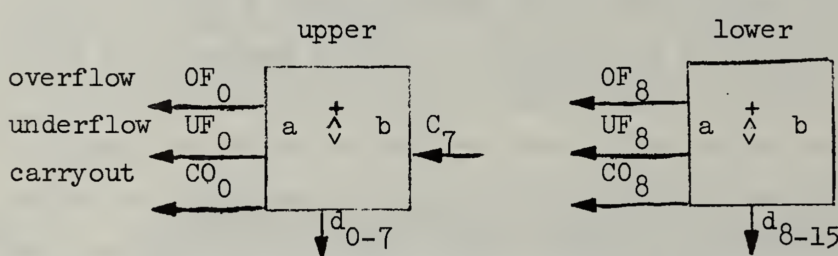
Two possible methods exist for generating the x and y voltages across the length and width of the tablet. The first method is to attack diodes with series resistors several places along each edge of the resistance sheet. The series resistance can then be varied to compensate for contact resistance and for forward-biased diode resistance to insure equipotential lines which are parallel to the edge. If five contact points are applied along each edge, the effective area of the sheet is reduced by a strip along each edge which is .75 times the distance between contact points. The second method which could be used would be to lay an arc*

* A circular arc can be used because it was shown that if its radius is r/R (where r is the resistance per square of the sheet and R is the resistance per unit distance of the linear, are shaped material) then the equipotential lines are straight and parallel.



of a material which has a low linear resistance (1 to 10 ohms/in.) along each edge of the tablet. The effective area of the tablet is reduced by the area of the tablet which is outside the arcs. The first method mentioned will be used on the first resistance tablet which is built. The second method mentioned would be desirable if arcs of a material with a low resistivity could be placed on the glass during the coating process.

3.4.3 General Description of Arithmetic Unit (R. Hostovsky)



Sixteen bit arithmetic unit shown as two eight-bit sections.

Generation of 3 bit Code for Incremental Display

The micro order APV has been designed for vector plotting. Small format Cartesian vector plotting uses the upper and lower parts of the ALU separately. Two's complement representation is implied here. Register X should be incremented if the upper ALU overflows (upper overflow OF_0 is the logical function $\bar{a}_0\ \bar{b}_0\ c_0$ where $a_0\ b_0$ are sign digits of the operands ("0" = positive) and c_0 is the carry into the most significant stage of the adder;

Register X should be decremented if upper ALU underflows (underflow UF_0 is $a_0\ b_0\ \bar{c}_0$);

Register Y should be incremented if OF_8;

Register Y should be decremented if UF_8.

Also the VP operation preserves the sign of an addition on overflow or underflow, i.e., do remains "0" after two positive numbers overflow the adder. Normally, overflow results in a negative representation of the "sum" (do = 1).

Thus, the upper and lower ALU add positive operands module 128 and the number of whole multiples of 128 is generating the incremental code.

ΔX and ΔY are 7 bits each, and they say by how much X and Y should be incremented, respectively.

For positive ΔX and ΔY overflows OF_0 and OF_8 will be generated at rates proportional to ΔX and ΔY, respectively, i.e., the original X and Y should be incremented proportionally to ΔX and ΔY. After 64 cycles there would be exactly ΔX overflows from the upper ALU and exactly ΔY overflows from the lower ALU. By cycle I mean the repetative addition of ΔX and ΔY to the former sum.

After each cycle OF_0 UF_0 OF_8 UF_8 are checked and the three-bit code is generated accordingly:

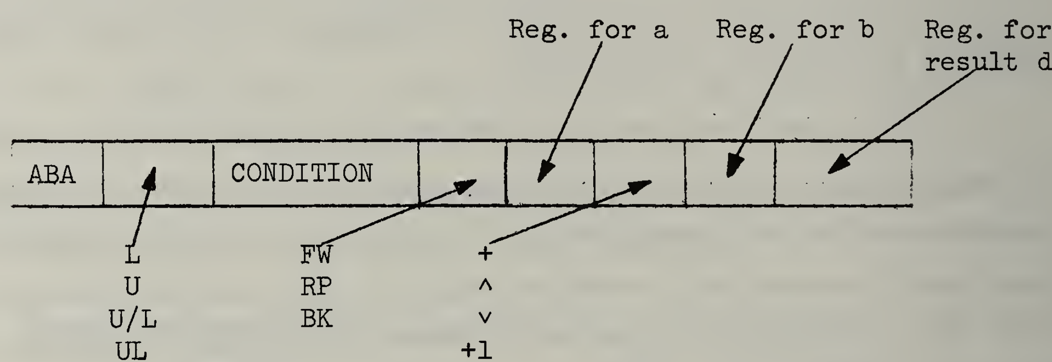
OF_0	UF_0	OF_8	UF_8	3 bit code
0	0	0	1	100
0	0	1	0	000
0	1	0	0	110
0	1	0	1	101
0	1	1	0	111
1	0	0	0	010
1	0	0	1	011
1	0	1	0	001

Micro-Code Outline (Arith/Logical Orders):

These orders add or perform a logical operation on the contents of two nominated registers and send the result to a third nominated register. Two's complement representation is used. Most of the machine registers are included so that many functions can be performed with these orders. For example, the core address being indexed by the contents of another register.

Another feature is the ability to repeat the same order or to backup and execute the previous one. This loop should be terminated upon satisfying some external condition.

a. ABA. The basic arithmetic/logical order combines operands a and b by addition or by digitwise logical operations.



b. ADR. Perform the same operation as ABA but in addition to it decrement a counter register R, so that in one micro order three operations are included: arith op, decrementing counter op, and repeat order. This goes on and on until R gets to some value, usually zero.

c. APV. As discussed in separate paragraphs, APV keeps adding ΔX and ΔY to themselves in order to cause overflow (underflow) and thus generating the three bit code for incremental display.

e.g. APV, U/L, RP, A+C $\rightarrow$ A

A register has been set at the beginning to zero. C contains ΔX (C_{0-6}) and ΔY (C_{8-14}).

d. ACO.

ACO		CONDITION	CONSTANT	DESTINATION

U
L
U/L
UL

The same as ABA but only puts the micro program constant field in some designated register.

Transfer Order

This order executes register to register transfer.

TRF		CONDITION	pairs of registers

U
L
U/L
UL

Block transfers will be performed by this order. Data will be read from core to some read-write register M and data will be transferred to nominated register. The core will be addressed by some register N which will advance upon moving data to and from M.

Set Mode Register

SMR		CONDITION			16 bits

SET
RESET

Bits of register S are set independently.

These bits are available for the condition register.

3.5 Numerical Analysis, Library Development (L. Fosdick)

The first of a series of quasi-interactive programs for short numerical computations was designed, and programming is nearly completed. This program allows the user to easily perform matrix operations, including inversion and solution of linear systems of equations. It is called a quasi-interactive system because the nature of the interaction is rather limited: the program requests data, for instance, "Please type the value of A(1,1)", but it does not converse with the user during, say, the inversion calculation if an ill-conditioned system is detected. In general, the conversation between the system and the user is only for fixing certain parameters necessary for calling a standard library subroutine. Once the subroutine has been called no further conversation is permitted for that job. After this program has been put into operation, a similar program for numerical quadrature will be constructed.

Trouble with library subroutines that has been detected after installation of a new release of the operating system has stimulated us to construct a master library subroutine checker. This checker is now complete except for about six library routines which are not checked. The checker is designed to call every library routine (excluding FORTRAN supplied functions) three times using three different parameter sets and checks the results against known results at several adjustable thresholds of error. The number three is arbitrary and may be any other number. If there is an error exceeding threshold the word NO is printed, otherwise, O K is printed. Thus there is only a single line of output for each parameter set tested. It is hoped that this system will allow us to catch most errors caused by system changes or accidental destruction of our library.

Systematic timing of all of our library subroutines, except four subroutines, has been completed. This information is now being incorporated in the short form library routine write-ups.

3.6 Computer Maintenance and Construction (C. Carter, H. Lopeman, R. Miller)

Check out of the PDP-8 disk channel was concluded this quarter. The disk is capable of transferring 1360 word blocks to and from the memory with reliable results. No test has been made on the reliability of information when retrieved after an interval of more than 30 minutes, as of this time.

Transfers are accomplished by the use of four IOT instructions:

IOT 6551 gates memory block starting address to the disk address register from the accumulator.

IOT 6562 gates control and disk block address information from the accumulator to the disk registers.

IOT 6561 skips on disk done.

IOT 6552 clears disk done.

Control word format is:

ACC. BITS	0123	45678	9	10	11
	DISK SECTOR	DISK TRACK	TR. DIR.	EXTENDED	ADDRESS
	1-12	0-15	1=READ 0=WRITE	2	1

IOT's 6551 and 6552 may be combined as microinstruction 6553 which transfers the address to the disk and clears disk done.

EXAMPLE PROGRAM FOR DISK READ AROUND

```
          50-0200  STARTING ADDRESS OF DISK WRITE BLOCK
          51-0410  WRITE CONTROL WORD (SECTOR 1, TRACK 1, TR. DIR. "0")
          52-0200  STARTING ADDRESS OF DISK READ BLOCK
          53-0415  READ CONTROL WORD (SECTOR 1, TRACK 1, TR. DIR. "1"
                   EXTENDED MEM. BANK 1)
START
         100-7200  CLEAR ACC.
         101-1050  TAD 50 LOAD LOC 50 TO ACC.
         102-6553  IOT TRANSFER ACC. TO DISK, CLEAR DISK DONE
         103-7200  CLEAR ACC.
         104-1051  TAD 51, LOC 51 TO ACC.
         105-6562  IOT TRANSFER ACC. TO DISK
         106-6561  SKIP ON DISK DONE
         107-5105  JMP-1
         110-7200  CLEAR ACC.
         111-1052  TAD 52 TO ACC.
         112-6553  IOT TRANSFER ACC. TO DISK, CLEAR DISK DONE
         113-7200  CLEAR ACC.
```

```
114-1053  TAD 53 TO ACC.
115-6562  IOT TRANSFER ACC. TO DISK
116-6561  SKIP ON DISK DONE
117-5116  JMP-1
120-5100  JMP TO START
121-
122-
```

The disk at this time appears to have sustained a massive head crash. The clock tracks have been wiped out, and the whole disk unit has been shipped back to the manufacturer for repair.

An interface to Illiac III from the PDP-8 has been designed, built, and checked out. The cabling is in and final testing will be performed in the near future.

The graphic input terminal tracking cross generator logic has been built and wired in. It is currently in a check out state.

4. ILLINOIS PATTERN RECOGNITION COMPUTER: ILLIAC III (Supported by Contract AT(11-1)-1018 with the U.S. Atomic Energy Commission)

Illiac III is an experimental computer being designed and constructed by the Department of Computer Science as a first instrument to explore the potentialities of high speed image processing. Besides providing normal computational facilities, the machine includes a parallel processor for visual data processing (the Pattern Articulation Unit) and an extensive complement of visual input/output equipment.

4.1 Outline of Illiac III Programming Developments

The section numbers of the outline below correspond to section headings of this Progress Report.

4.2 Operating System

- 4.2.1 Data Acquisition and Display
  - 4.2.1.1 Image Processing Package (IP^2)
  - 4.2.1.2 Telecommunication Processing Package (TP^2)
- 4.2.2 Data Segmentation
- 4.2.3 Multiprocessing and Interrupt Handling

4.3 Translators

- 4.3.1 IBAL Assembler Language
- 4.3.2 PL/1

4.4 Experimental Image Recognition Procedures

- 4.4.1 Graph Transformation Grammars
- 4.4.2 Experimental Model

4.2 Operating System

4.2.1 Data Acquisition and Display

4.2.1.1 Image Processing Package (IP^2)

As the basis for developing an adequate language for image description and recognition, a review of graphical languages and recognition models has been undertaken. Representative languages have been evaluated with respect to criteria we have established. No language or combination of languages satisfies all of our criteria.

Work is in progress on a language as powerful as existing languages and with additional capability of expressing varied modes of concept formation.

John C. Schwebel

4.2.1.2 Telecommunication Processing Package (TP^2)

A task is the smallest unit operation which the computer may be requested to execute subject to a given deadline for completion. The command or instruction which causes the task to be executed is called a task directive. Accordingly task directives are commands or messages sent from the user's console to the computer to specify the next gross action the computer is to take.

The set of permissible directives defines a programming language — one that is directly interpreted by the supervisor program. All other languages available to the computer system are considered subservient to this Task Directive Language. For this reason the latter language has been kept as simple and transparent as possible so as not to needlessly burden the user.

A description of the language can be found in File No. 782, "Task Directive Language for the Illiac III Computer" by Linda M. Katoh, December 9, 1968.

4.2.2 Data Segmentation

In spite of draft notices, etc. this quarter, work continued on the problems of segmentation. In particular, a tentative design has been reached which appears well suited to multi-processing and/or multiprocessing. Work presently centers on specifying the exact effect of this design on the TP control design.

J. D. Wick

4.2.3 Multiprocessing and Interrupt Handling

During the past quarter, a scheme has been formulated to facilitate interrupt handling in multiprocessing environments. In particular, the hardware associates interrupts with the generating process and not the executing processor. In general this feature allows processors to be treated as an anonymous pool and should greatly reduce the necessity for inter-processor communication.

A paper setting forth the details as currently envisioned is now in progress.

R. M. Lansford

4.3 Translators

4.3.1 IBAL (Assembler Language)

Work was continued on Pass 1 of the IBAL translator during the quarter. The source program's block structure is incorporated into the structure and identifer table by including the block number as an attribute with the identifier's entry.

Subscript bounds are handled more completely. The string defining each bound is recognized and placed into the output string in the same manner as other program elements, with a tag which is also stored in the structure table along with the subscript's entry. The bound will be evaluated (or if it is variable, code will be produced to evaluate it) during a later pass.

Initial constants for identifiers are recognized as to type and pushed into a stack in more easily evaluatable form, and a pointer to the constant is placed in the structure table. Some diagnostic messages regarding syntax of initial constants have been added.

It is anticipated that the only further major additions that will have to be made to Pass 1 are suitable indication of alternate access structures and conditional format items.

S. T. Baird

4.3.2 PL/1

During this quarter a set of nearly 450 productions for PL/1 was completed. In addition, the set of actions necessary to build tables during the Declaration pass was finished.

R. M. Lansford

4.4 Experimental Image Recognition Procedures

4.4.1 Graph Transformation Grammars

Work continued on the study of the precedence of image processing relations with respect to some specific graph transformations. The concept of a reversible transformation, i.e. an information lossless transformation, was defined. This concept was used to determine the relative precedence of our set of common image processing relations. One useful result is that the use of an information lossless transformation in a recognition procedure can be justified without regard to the particular application area or constituents involved.

4.4.2 Experimental Model

Experiments were conducted using a model in which an observer can sequentially view subparts of a scene, selectively choosing at each step the location of the next subpart in the sequence.

Line drawings of geometric figures, and slides from microscopic pathology were used as subject areas.

Strategies used by the subjects in selecting subpart placement were readily apparent in the highly structured area of geometric drawings but were not apparent in the examination of the random configuration of cells in human tissue. This may be due to the fact that continuous areas of the scene were the only subparts allowed in the selection process.

Further work with this model awaits the development of an adequate language to describe the structural and procedural aspects of recogntion.

John C. Schwebel

4.5 Outline of Illiac III Computer System

The section numbers of the outline below correspond to section headings of this Progress Report.

4.6 Central System

- 4.6.1 Taxicrinic Processors
- 4.6.2 Fast Core Storage Modules
- 4.6.3 Arithmetic Units
- 4.6.4 Interrupt Unit
- 4.6.5 Pattern Articulation Unit
- 4.6.6 Exchange Net and Exchange Net Simulator
- 4.6.7 Status of the Main Frame Assembly

4.7 I/O System

- 4.7.1 I/O Processors
- 4.7.2 Channel Interface Units

4.8 Peripheral System

- 4.8.1 Secondary Storage System
- 4.8.2 Scan/Display System
  - 4.8.2.1 Scanner-Monitor-Video Control
  - 4.8.2.2 Prototype Scanner
- 4.8.3 Intermachine Link to DEC PDP-8 and 338 Display
- 4.8.4 Low Speed Terminal Network
  - 4.8.4.1 Low Speed Communications Net
  - 4.8.4.2 Low Speed Buffers
  - 4.8.4.3 Low Speed Terminals
    - 4.8.4.3.1 Monitor Selectric Typewriters (5)
    - 4.8.4.3.2 Monitor Magnetic Tape Module (5)
    - 4.8.4.3.3 Mod 33 ASR Teletype Sets
    - 4.8.4.3.4 Analog Instruments

4.9 Power Distribution

4.9.1 Primary DC Power Supplies

4.9.2 Power Distribution System

4.9.3 Control of Power Distribution Systems

4.9.4 A.C. Power Distribution System

4.10 Unassigned Equipment Pool

4.10.1 Circuit Card Inventory

4.10.2 Test Equipment Additions: Commercial

4.10.3 Test Equipment Additions: Custom-Designed

4.11 Documentation

4.11.1 Engineering Manual

4.11.2 Circuit Books

4.11.3 Logic Book

4.11.4 Wiring Tables

4.11.5 Opto/Mech. Design

4.12 Circuit Research and Development

4.6 Central System

4.6.1 Taxicrinic Processors

4.6.1.1 Documentation

Progress continues on the TP manual. The interrupt system description is currently being integrated into all sections of the manual.

4.6.1.2 Logical Design

The TP interrupt system is currently being integrated into the overall TP design. The various modifications to the control sequence flowcharts have been made. The basic operation of the Interrupt Control sequencing has been worked out and is currently being implemented.

4.6.1.3 Hardware and Wiring

One TP lower bay has been returned from the wiring vendor.

B. J. Nordmann

4.6.2 Fast Core Storage Modules

This quarter each Fabri-Tek core was checked out, on line, using SPEW (see 4.7.6). Some discrepancies were found as follows:

Core 66700	Core 66701
1) 1 wiring error	1) 1 bad IC on parity adder board
2) 1 bad Burndy contact	2) 1 bad IC on Y-driver adder board
	3) 1 bad etch on receiver adder board

The first of two sets of modifications to the Fabri-Tek cores has been completed and checked out. The remaining set of modifications should be completed during February 1969. These modifications are required to make the core modules compatable with the Exchange Net.

Several on-line, error free runs were completed with each core and SPEU. During these runs, both data and parity were checked. Programmable data patterns were sequenced, by SPEU, such that every data pattern was used in every core address. A test at the byte read/write control bits was also incorporated in the sequencing.

S. P. Krabbe

4.6.3 Arithmetic Units

4.6.3.1 Logical Design

During the last quarter logical design of the A.U. control has continued. The control flowcharts, in a form which may be routinely mapped into logic, have been completed. Some refinement of these flowcharts takes place as the actual logic is specified.

As the detailed design of control logic continues, we are attempting to document the steps required in the transformation of flowcharts into logic (T.I. 7400 series). Such documentation will be of value to other designers and serve as a basis for the implementation of a computer program to perform this task.

4.6.3.2 Hardware and Wiring

Three configurations of control point boards have been specified and prototypes have been constructed. The three boards represent a varying tradeoff between flexibility and packing density.

Wiring tables for the 475 cards (per AU) of the processing hardware is now complete. Revisions made on the work prints during the course of wiring table preparation are now being transferred to the original drawings. The wiring tables are then being checked against the revised drawings.

D. E. Atkins

4.6.4 Interrupt Unit

No additional work was done on the design of the Interrupt Unit because of commitments of higher priority.

4.6.5 Pattern Articulation Unit

4.6.5.1 Logical Design

No work has been done on the design due to other commitments. It is hoped the design will be completed near the end of next quarter.

4.6.5.2 Documentation

No positive progress made during the past quarter.

4.6.5.3 Hardware and Wiring

The Display System has been debugged and all documentation brought up to date. The 32 Local Control cards are being fabricated.

Power wiring to Section 3 (Iterative Array) is partially complete. It is anticipated that a portion of the IA will be checked out during the next quarter.

R. T. Borovec

4.6.6 Exchange Net and Exchange Net Simulator

This quarter, wiring lists for the Exchange Net were completed and wiring was started.

SPEU (Simulator for Processor, Exchange, or Unit) checkout is complete. This last quarter, SPEU was used for the online checkout of our two Fabri-Tek core memory systems (see 4.7.2).

S. P. Krabbe

4.6.7 Status of the Mainframe Assembly

The following list of items describe mainframe progress during the last quarter.

1) Logical and power wiring completed on sections 1 and 3 lower (Iterative Array). These two half sections are now installed in the mainframe. This completes the Iterative Array.

2) Logical and power wiring completed on Section 6 (Taxicrinic Processor minus control). This section is now installed in the mainframe.

3) A request for bids on a false floor, for DCL 223, that overlaps the existing false floor in DCL 280, was issued this quarter.

4) DC power distribution has been started inside the mainframe of Illiac III.

This quarter, the following wiring lists were made using POOL I.

Subsection	Section #	Number of Wires	Comments
Taxicrinic Processor	6	9622	
Simplified Scanner (Row A)	33B	926	
Simplified Scanner (Row B)	33B	257	
Exchange Net	13	5377	
Exchange Net	14	7536	
IA Display	1 and 3	1552	No revision

S. P. Krabbe

4.7 I/O System

4.7.1 I/O Processor

The subsystem of the I/O Processor responsible for data transmission (the request service unit) has been reexamined this quarter. A preliminary design study was completed by John Hayes in 1967, but layout work was postponed because of the Taxicrinic Processor and Scanner-Monitor-Video Controller design activities. The Hayes design has been retained in form but has been considerably simplified.

A reorganization of that part of the IOP responsible for executing device independent commands is also underway. Besides certain simplifications in design, the main motivation of this latter work is to permit easy transplantation of modules of the Taxicrinic Processor control. For example the addressing structure of both boxes is common.

Present logical design activity is centered on realizing a first interface to core through a Channel Interface Unit, employing that subset of the IOP which is essential to the task.

B.H. McCormick

4.7.2 Channel Interface Unit (CIU)

Work in this area has been necessarily divided into three groups:

1) Documentation (4.8.2.1)
2) Logic Implementation (4.8.2.2)
3) Construction (4.8.2.3)

4.7.2.1 Documentation

While documentation is a continuing process through all work groups, in this instance we are referring to the processes of:

1) Complete perusal of earlier data for the purpose of separating documentation of historical interest only from that which can be used for implementation of the CIU as it will eventually exist. (Complete)

2) Upgrading of flowcharts, written data and logic diagrams for correction, streamlining, practicality, and ease of use. Since signal requirements, sequences and processes are now firm, this work is proceeding at a reasonable rate. (approximately 25% complete)

4.7.2.2 Logic Implementation

This consists solely of construction of logic diagrams for implementing processes designated by flowcharts and current written data.

Since firming up of processes designated in paragraph 4.8.2.1, and design, layout and acceptance of a firm Control Point circuit and printed circuit boards, this particular work group is proceeding rather faster than the documentation (approximately 60 to 70% complete)

4.7.2.3 Construction

This includes PCB selection, layout, generation of wiring lists; actual wiring; and mechanical construction. Work in this area must wait until documentation and logic implementation are nearly complete.

Therefore no work has been done on this particular group, with the exception of mechanical construction. Card drawers and racks were constructed some time in the past and are essentially complete.

J. V. Wenta

4.8 Peripheral System

4.8.1 Secondary Storage System

No developments are anticipated here until after October 1969 due to budgetary constraints.

4.8.2 Scan/Display System

4.8.2.1 Scanner-Monitor-Video Control

A flowchart description of the controller reflecting modularity and control point implementation is nearing completion.

Preliminary flowcharts have been drawn for the section of the control unit which will provide the standard IBM System/360 I/O Interface signals as defined by the appropriate IBM SRL. Several poorly defined sequences are under further investigation.

For the basic structure complete detailed drawings have been drawn for the counter structure. The register structure, the equality function, the theta function, and the input/output buffers have been completely specified. The remaining four functions are currently being specified.

A paper entitled "Scan-Display System of the Illiac III Computer" was accepted for presentation at the Spring Joint Computer Conference in Boston in May 1969. This paper describes the integrated system used for image acquisition and display that is under the supervision of the Illiac III Scanner-Monitor-Video controller. The major goal of the paper is to abstract the system parameters and to develop the relations among them.

L. Dunn
V. Tareski

4.8.2.2 Prototype Scanner

A simplified scanner has been built as a means of testing various logic and analog portions of the final SMV system. The simplified scanner scans a 512 x 512 raster in one of 64 selected areas of the object film, packs the gray level information into 32 bit words and stores this information in one of the Fabri-Tek memories. The same logic permits the stored information to be extracted, decoded and displayed on a monitor. The system is operational except for adjustments to be made in electrical and optical focusing, in timing of the DAC circuits and in photomultiplier signal conditioning filters.

J. L. Divilbiss

4.8.3 Intermachine Link to DEC PDP-8 and 338 Display

The PDP-8 link logic has been designed and debugged. One small error still occurs, namely the parity error flag is set whenever the space bar on the teletype is hit.

A series of programs, GRITS (General Interface Testing System), was written to facilitate checkout; a second series of programs is currently being developed to help in the checkout of the Herative array.

The work on a master system program which would allow the engineers to write code in a simple macro-like language has been temporarily halted due to the need of the project's programmer in another area.

R. T. Borovec

4.8.4 Low Speed Terminal Network

4.8.4.1 Low Speed Communications Net

No progress made during this quarter.

4.8.4.2 Low Speed Buffer

4.8.4.2.1 Low Speed Buffer Control

Because of changes from the original operating specifications, the design of this unit is being completely re-evaluated. The packaging and physical location of this unit is also being considered.

4.8.4.2.2 Buffer Memory

This unit has satisfactorily completed its initial testing phase and slight modifications are being made based upon the results of these tests.

4.8.4.3 Low Speed Terminal

4.8.4.3.1 Monitor Selectric Typewriters

There has been no changes in the status of these units during the past quarter.

4.8.4.3.2 Monitor Magnetic Tape Modules

The type of format and associated hardware required for these units is still being investigated.

4.8.4.3.3 Teletype Sets

(1) Mod 33 Teletype unit has been interfaced with the DEC PDP-8 to provide I/O with Illiac III.

4.8.4.3.4 Analog Instruments

No further design has been completed in this area due to prior design commitments.

R. G. Martin

4.9 Power Distribution

4.9 .1 Primary D.C. Power Supplies

All fire damaged power supplies have been repaired, tested and are now operational.

While it is evident that additional primary supplies will be needed as future current requirements are realized, initial sections of the machine will be adequately supplied by the units now on hand.

All primary distribution cables have been laid and connected to the distribution centers.

4. 9.2 Power Distribution System

4.. 9.2.1 Primary D.C. Distribution Center (Room 223)

The connection of modine channels, capacitors, buss bars and associated cables for (1) PAU wing (Section 3) has been initiated and will be completed by the next quarter.

4. 9.3 Control of Power Distribution System

The wiring of the card racks used to control the power distribution to section 3 of the mainframe has been started this past quarter.

4. 9.4 AC Power Distribution System

There have been no further additions or changes in this section during the past quarter.

R. G. Martin

4.10 Unassigned Equipment Pool

4.10.1 Circuit Card Inventory

The printed circuit cards required for the minimum version of the Illiac III have been allocated and are stored per section of the machine. Using this procedure, there should be no confusion as to the location and stock disposition of a particular type of card required for a given section of the machine.

In addition to the assigned printed circuit cards, there is approximately 4000 cards in open stock which will be used for prototype and new design testing.

4.10.2 Test Equipment Additions: Commercial

No additional commercial equipment obtained during this quarter.

4.10.3 Test Equipment Addition: Custom-Design

A storage core simulation tester has been constructed for the checkout of the Fabri-Tek fast core storage modules.

R. G. Martin

4.11 Documentation

4.11.1 Engineering Manual

This manual has been updated with the latest additions of new circuit drawings and engineering comments.

R. G. Martin

4.11.2 Circuit Books

Work was completed for release on thirty circuit cards. Work was either new circuit or variation of existing circuits. In addition eight change notices were processed on circuitry.

4.11.3 Logic Book

Work was completed for the EU and TM intermediate drawings. The TP logic was revised twice during this time. Power logic documentation is half completed.

4.11.4 Wiring Tables

No changes on the wiring tables were reported.

J. Otten

4.11.5 Opto/Mech. Design

From October 1, 1968, through December 31, 1968, twenty four (24) new detail, assembly and layout drawings have been made.

Four (4) additional drawings have been processed according to engineering change orders.

Six (6) work orders have been placed with our machine shop.

Opto/mechanical design of the manually-controlled microscope slide scanner was completed during this quarter.

S. Zundo

4.12 Circuit Research and Development

This quarter saw the completion (in final printed circuit form) of several analog and IC circuits.

1018-300-00 1018-306-00	These two card types make up a digital to analog conversion system with speed and stability superior to commercially available units. Much time was spent in developing these circuits but our efforts seem well rewarded.
1018-301-00	Digital control of CRT beam brightness plus phosphor protection in the event of logic failure.
1018-302-00	A thirty section tapped delay line with receivers. Ten nanoseconds per tap.
1018-303-00 1018-307-00	Sine and cosine function generators for use with the diquadripole lens.
1018-304-00 1018-305-00	Integrated circuit control point logic and associated delay modeling circuit.
1018-308-00	High speed clock (4 NS risetime) for use with DAC buffers.
1018-309-00	A digitally controlled analog switch. Logic input determines which of four analog inputs (+sine, -sine, +cosine, -cosine) appears on output.
1018-310-00	A logarithmic amplifier used in photomultiplier signal conditioning.

J. L. Divilbiss

4.13 Bibliography

During the past quarter the following reports and file numbers were issued:

Reports

Report No. 291	Flowerdew, Stanley J., "LEFT: A Language for Editing and Formatting Text, October 30, 1968.

File Numbers

File No. 780	Flowerdew, Stanley J., "Automatic Karyotyping: A Statement of the Problem and Summary of Image Processing Procedures", November 6, 1968.
File No. 782	Katoh, Linda M., "Task Directive Language for the Illiac III Computer", December 9, 1968.
File No. 550-118	Krabbe, S. Paul, "Specifications for Termi-Point Wiring of Section 6 (Taxicrinic Processing) of the Illiac III, November 29, 1969.
File No. 550-119	Krabbe, S. Paul, "Specifications for Wire Wrap of Section 13 (Exchange Net), November 6, 1968.
File No. 550-120	Krabbe, S. Paul, "Specifications for a 9" False Floor for DCL 223 that Overlaps the Existing False Floor in DCL 280, November 5, 1968.
File No. 550-121	Krabbe, S. Paul, "Specifications for Wire Wrap of Section 14", November 5, 1968.

Other Publications

Atkins, D.E., "Higher Radix Division Using Estimates of the Divisor and Partial Remainders," IEEE Transaction on Computers. Vol. C-17, No. 10, (October 1968), pp. 925-934.

4.14 Illiac III Staff[1]

Senior Staff

Dr. James L. Divilbiss - Principal Research Engineer
Dr. Robert M. Lansford - Research Associate
Professor Bruce H. McCormick - Principal Investigator
Professor Sylvian R. Ray
Professor K. C. Smith - Consultant

Professional Staff

Robert C. Amendola
Richard T. Borovec
Lawrence A. Dunn
Mrs. Linda M. Katoh
Ronald G. Martin

Electronic Engineering Assistants

David L. Foster
Thomas E. Kingery
S. Paul Krabbe
Edmund Pelg
Frank P. Serio
Joseph V. Wenta

Electronic Technician II

John L. Fileccia
Robert G. Thomson

Electronic Technician I

Eugene E. Althaus
George T. Lewis
William A. Miller
Bennie D. Williamson

Senior Laboratory Mechanic

Ralph M. Rivera

Engineers

*Daniel E. Atkins
*Thomas T. Chen
*Sergio Franco
*Lakshmi N. Goyal
*Richard P. Harms
*Bernard J. Nordmann
*Val G. Tareski

Programmers

*Stuart T. Baird
*John C. Schwebel
*John Wick

Drafting

John Otten
Stanislavs Zundo

Secretarial

Mrs. Donna J. Stutz

---

[1]Not including hourly employees
*Research Assistants

5. ILLIAC IV

This work was supported in part by the Department of Computer Science, University of Illinois at Urbana-Champaign, Urbana, Illinois, and in part by the Advanced Research Projects Agency as administered by the Rome Air Development Center, under Contract No. US AF 30(602)4144.

REPORT SUMMARY

The following major problems became apparent during this quarter.

Semiconductor Devices

As a result of detailed investigation and continuing pressure by Burroughs, information was obtained that Texas Instruments (TI) was in considerable difficulty developing the MSI devices for ILLIAC IV. The problems TI encountered fell into the categories of testing equipment and obtaining multilayer substraits. The ILLIAC IV MSI devices are 80 pin devices and the only available tester is a 44 pin tester. TI attempted to purchase other testers but the subcontractors were not able to produce. TI had contracted with American Lava to supply ceramic substraits. For a variety of reasons American Lava was not able to produce. As a result of the above problems, the PE contract with TI was terminated and an investigation began on replacing the MSI devices. The only solid technology base available to the ILLIAC IV Project were DIL's (Dual In Line). The decision was made to use the DIL's being produced by TI as part of the ILLIAC IV contract. This decision impacts the processing element as the PE was being designed with both MSI devices and DIL's. The Control Unit (CU) had not been designed with MSI devices. Burroughs is in the process of developing a new mechanical configuration for the processing element and making the appropriate modifications to the design to incorporate the details.

Memories Systems

A review of Burroughs' plans and projected cost in producing the thin film memories created a situation whereby the University felt obligated to require Burroughs to solicit bids from industry to provide backup for the thin film memories program. Burroughs issued a request for quotation to memory suppliers, including all the major semiconductor manufacturers. The responses were from semiconductor manufacturers and not a single thin film memory system was bid. In addition, the firm fixed price bids were approximately one-third (1/3) the cost of the proposed thin film system. As a result, the thin film memory program has been terminated at Burroughs and a procurement activity started for semiconductor memories. At this time, Motorola, TI, and Fairchild are providing quotations on ILLIAC IV memories.

Impact on Cost, Schedule and Performance

The retrenchment to a DIL technology will have impact on cost. This impact is in the process of being estimated by Burroughs. Schedule impact is estimated to be in the neighborhood of six (6) months which will provide the first quadrant of ILLIAC IV delivered to the University in June of 1970. The implementation with DIL's will result in a volume increase in the ILLIAC IV. It is estimated that an additional 15 feet will be required per quadrant row. As a result of the volumetric increase a slowdown in the clock system must occur. It is estimated the clock will slow down to about 60 nsec.

Plans for Next Quarter

The major efforts during the next quarter will be to complete the PE design using DIL's to award a contract for the PE memories and to provide detailed information of the above problems on cost, schedule and performance. There have been no significant personnel changes during the last quarter. Current funding is adequate to carry the Project through mid-March.

5.1 Logic Debugging and Diagnostics

5.1.1 Simulation and Debugging of PE Logic

The logic simulation and debugging of the Processing Element design have been in progress during this quarter. Two programs were written, in this period, to assist in the procedure of debugging - one for counting the logic levels and the other for identifying Boolean equations of major PE signals.

5.1.1.1 Improvement of the Logic Simulator

A few changes were made in the logic simulator program to reduce its running time. The simulator, as well as the Assembled Code Translator, received some changes caused by the altered, sign bit convention.

5.1.1.2 Level Counting Program

A modified version of the logic simulation program, which counts the logic levels associated with the control logic of the PE, was written for the purpose of estimating the time delay and debugging the control logic design. The procedure call statements associated with the control logic were selected out of the logic simulator body, and the parameter type was changed from Boolean variable to integer variable. The new procedures for calculating the logical levels were for all package types, and some additional coding was done to complete the level counting program. It was developed according to a request from the Burroughs' PE designer.

5.1.1.3 Equation Generator Programs

The DEG (DIL Equation Generator) writes an equation for dual-in-line packages in the PE. For the latch, DIL type 8, only the signal names are listed. The equation is written using the signal names associated with the package. Input to the program is the wire list which is scanned to separate DIL from MSI packages. The output of this program serves as input to the PE equation generator and can be used as a debugging aid.

The PEG (PE Equation Generator) develops an equation for P-signals in the PE control logic in terms of primitive inputs. A primitive input is a signal external to the PE or it is the output of a latch or an MSI package. The program is an extension of DEG in that it steps through the list of DIL equations generated in DEG and replaces any non-primitive input signals with the equation for the signal in terms of primitive inputs. The output of this program can be used as a debugging aid for the control logic of the PE.

5.1.1.4 Debugging

The simulator program was executed on test inputs from three sources:

(a) card decks of single cases manually defined,

(b) PE Exercisor test routines (from Burroughs) translated by the Assembled Code Translator, and

(c) automatic extraction of microsequences from POSFILE (PE instruction timing sheets).

Mechanization of basic arithmetic instructions, including logical addition and subtraction, fixed point multiplication, and exponent arithmetic, was debugged with card decks. PEX routines for Boolean and shift instructions, mode register instructions, eight-bit mode arithmetic, and some other instructions have also been tested on the simulator. The POSFILE approach was used to debug normalization and floating-point arithmetic instructions. Approximately 50 errors were found by the simulator by the end of this quarter. The major instructions to be debugged in the next quarter include division and some arithmetic instructions with variants.

5.1.2 CU Logic Simulator System

5.1.2.1 Overview

Specifications have been drawn up for two simulator systems to assist in the logical design and debugging of the Control Unit. These are the CU Card Logic Simulator System and the CU Section Logic Simulator System. Although the construction of these simulators will be similar to

that of the PE Simulator described in the previous QPR, several improvements have been included which should yield much greater simulation efficiency. Among these are a capability for parallel simulation of many test cases at the same time and a new simulation control language to permit simple specification of complex simulations.

5.1.2.2 CU Card Logic Simulator System

The purpose of the CU Card Logic Simulator System is to assist in debugging CU printed circuit card designs and to calculate expected responses for various test cases. In addition, it will be used to develop card simulation procedures for inclusion into the CU Section Logic Simulator System. The card simulator system will also incorporate a capability for the modification of the CU Card Logic Simulator to simulate the effect of logic failures. This will help in developing input patterns to be used for production tests and off-line diagnostics of the CU cards.

The CU Card Logic Simulator System will consist of four major groups of programs: the simulator generator group, the data preparation group, the simulator itself, and the results display group. The simulator generator group of programs are similar to those for the PE simulator. They consist of programs which reduce the CU card to a directed graph, make preliminary level assignments, detect loops, perform final level assignments, and then generate the actual simulator body. Two improvements for increased simulation efficiency have been added: logical simulation of the dual-inline packages is performed by in-line coding rather than by time-consuming procedure calls, and loop members are simulated along an optimal (Hamiltonian) path (if one exists) rather than in random order as in the PE simulator.

Much of the CU card logic simulator is similar to the PE logic simulator. However, a new, more flexible simulator input/output system has been designed. Also, the simulator can operate in modes which are not physically realizable but nevertheless are of potential use in logic debugging (e.g., registers can be prevented from changing state and can be preset to any desired contents). Further, the simulator has been designed with two additional goals in mind: first, to provide parallel simulation of up to 47 independent test cases by taking advantage of the fact that the

B5500 performs Boolean operations on full (47-bit) data words, and second, to permit simulation of logic failures by simple modifications to the simulator body.

A new simulation test control language, TESLA, has been developed. This test language permits simple description of desired simulations. The language is very general; its later use with the CU Section Logic Simulator System is anticipated. One unusual feature of TESLA is the incorporation of language elements for simultaneous generation and control of many simulations, thus complementing the parallel simulation capability of the simulator itself.

Since testing and fault diagnosis for the actual CU cards will be performed on the PEX (with a CU card tester adapter), it is desirable to be able to simulate these tests. This will be done by providing a method for translating PEX programs into simulator input.

On December 2nd, a meeting was held with Burroughs designers at Paoli. The proposed card simulator system was reviewed, and several helpful changes and improvements (particularly to TESLA) were suggested. Since that meeting, the design of the CU Card Logic Simulator System has essentially been completed, and coding of the necessary routines has begun. CU cards will be simulated as soon as the necessary netlists become available, with precedence being given to cards from those sections of the CU which will be completed earliest.

5.1.2.3 CU Section Logic Simulator System

Preliminary specifications for the CU Section Logic Simulator System have been developed. It will simulate the logical performance of one entire CU section (FINST, ADVAST, ILA, MSU or TMU). The CU section simulator will incorporate the bodies of the card simulators as procedures. Actual parameters to these procedures will be variables corresponding to the backplane signals.

Each section will also have a functional simulator developed for it. This functional simulator will be identical to the (correctly functioning) CU section logic in external performance, but will hopefully be far simpler in internal construction and thus much more efficient. It will

be constructed, insofar as possible, from narrative descriptions of the desired section performance rather than from actual logic diagrams. Thus, it should be somewhat insulated from simple design errors and should provide a valuable performance standard and source of expected responses.

Since writing a correctly operating, functional simulator exhibiting the same full range of dynamic performance as the actual logic will be no simple task, a bootstrapping technique will be used to permit continuous interdependent upgrading and improvement of the five functional simulators until the desired level of performance is achieved.

An important advantage which will accrue from the development of the functional simulators will be the ability to use actual assembled ILLIAC IV code to test any (simulated) section of the CU. Functional simulation of the other sections will provide the necessary translation of input code into the actual commands and data which are sent to the section under test.

Because a correctly functioning card simulator system is a necessary preliminary to the development of the section simulators, only initial efforts at defining some of the more global aspects of the CU Section Logic Simulator System have thus far been possible. However, substantial portions of the section simulator system design should be completed during the next quarter as designer manpower becomes available and as operating experience with the CU Card Logic Simulator System provides valuable feedback to the design process.

5.1.3 PE Diagnostics Generation

5.1.3.1 Path Tests

The debugging of the expected response calculator for the path test was continued, and the corrections of several errors were made in this period. The path tests will be finalized after the debugging of the PE logic.

5.1.3.2 Combinational Tests

Due to the recent shift from the implementation of the PE in MSI packages to the DIL approach, the test patterns for combinational logic will be reworked in the next periods.

5.2 Design Automation

The major effort of the design automation group, during this quarter, was giving substantial programming support to Burroughs, Paoli, Pennsylvania. The Mohawk data set is continuing to be used for design automation production jobs.

A single board Post-Processor was written and debugged at the University. This Post-Processor, which has new and enlarged specifications, is capable of examining each wire on the PC Board. Work has also begun on a Post-Processor capable of handling the entire CU with inter-board wires.

5.3 Operating System

The operating system provides several services each of which is implemented by one or more program modules within the system. The user interface to the operating system is through the job parser which can be driven by any available input media such as tapes, card decks, user operated consoles, or other running B6500 programs.

The collection of programs and some of the data needed on the ILLIAC IV is performed within the B6500 by a program collector.

The scheduling of ILLIAC IV time is performed by three system modules: the disk file allocator, the data pre/post processor, and the job execution monitor. These three programs queue up requests for ILLIAC IV use and assign Disk IV space to them, move any files that are needed onto the disk before a run, save any files by moving them off the disk after a run, schedule the use of BIOM space, and allocate particular ILLIAC IV quadrants to the individual jobs.

There are two main ILLIAC IV resident modules, the loader and OS4. The loader loads program files on Disk IV into the array memory. OS4 provides the standard monitor functions and the I/O execution coordination routines needed by all users.

The running ILLIAC IV job requires at least two intercommunicating programs--the ILLIAC IV resident program and a B6500 resident job partner that coordinates B6500 actions with the ILLIAC IV program and provides all I/O support. Each job must have at least one job partner for all of its quadrants and each job may have as many job partners as it has separate quadrant code streams. A user may choose not to use the system-supplied job partner with the execution of his ILLIAC IV program and may write his own job partner.

The building of I/O descriptors, the initial recognition of interrupts, and the actual issuing of I/O commands to the hardware are functions of a set of routines in the B6500 MCP which are collectively known as the hardware supervisor. Since a user may write his own job partner,

the hardware supervisor also checks all of the I/O requests made by a job partner for validity before passing them onto the hardware.

Most of the modules named above are separately running B6500 ALGOL programs. They communicate with each other through the use of the in-core file facility provided by the B6500 MCP. The hardware supervisor routines (called "MCP intrinsics") are coded in ESPOL to facilitate both the issuing of I/O descriptors and the passing of interrupts to the correct job partner. The ILLIAC IV loader and OS4 are coded in the ILLIAC IV assembly language.

A preliminary version of OS4, the ILLIAC IV resident monitor, is being written and debugged. It soon will be published as a document.

5.4 Translator Writing System

5.4.1 Syntax Preprocessor (BNF → FPL)

During this quarter, the Floyd production and parser pseudo-order generator of the preprocessor were completely rewritten to implement improvements in the algorithm and effect the removal of empty productions and singly defined nonterminals by back-substitution.

The first stage of documentation, which describes the algorithm used in developing the syntax preprocessor, has been completed [1].

The first version of the syntax of Twinkle, a syntax specification language, was punched and partially debugged. During the next quarter, work on Twinkle will be stepped up since the syntax of Twinkle, with comments, will be the next stage of the documentation on the use of the syntax preprocessing portion of the TWS.

5.4.2 Parser (FPL)

The implementation of the improved BNF to FPL conversion algorithm necessitated changes in the parser interpreter. These changes were made and debugged which resulted in faster execution speeds.

The program CONVERT/TWS, which takes the table of parser pseudo-instructions generated by SYNPRØF/TWS and converts them to ALGOL, was written and debugged. Prior to the latest changes it achieved parsing times

equal to the scanning times. Early in the next quarter, it will be updated to convert the most recent output from SYNPRØF/TWS. A program is being contemplated which will automatically update the program CØNVERT/TWS to solve the problem of the long time lag between the improvements in the parsing algorithm and the correction of CØNVERT/TWS to handle them.

5.4.3 Semantics

During this quarter, the TWS-built ISL translator was completed and debugged. Consequently, the TWS system now has two complete ISL translators: (a) the "brute-force" ISL translator, and (b) the TWS-built ISL translator.

The advantages of (b) over (a) are basically two. First, the TWS-built ISL translator, having been implemented with the TWS system, is easily modifiable by anyone familiar with such a system. This is a very important feature, as it facilitates future extensions of ISL. Secondly, (a) implements a more sophisticated version of OSL than the basic ISL implemented in (b).

The main disadvantage of (b) over (a) is that (b) is considerably slower than (a), as could be expected, since (a) was very carefully optimized by hand. As the TWS system improves, however, it is hoped that the speed of (b) will approach that of (a), and (b) will become the only ISL translator. Tests are now being performed on the speed of the two translators.

Work is also continuing on the documentation of the language and by the end of next quarter a complete ISL manual should be available.

5.5 Compilers and Translators

5.5.1 Tranquil

During this quarter, work continued on the Pass 2 semantics for the Tranquil compiler. Progress on the Tranquil compiler has been delayed by the partial unavailability of the B5500. Documentation of Tranquil specifications and implementation techniques is now available in the form of three MS theses [2,3,4].

Routines which handle all storage allocation and compile time memory maintenance are fully coded and debugged. Some effort at improving the efficiency of the ALGOL coding of these routines has been initiated.

The complex substructure necessary for the analysis of assignment statements has been coded and debugged. Algorithms for code generation for assignment statements have been designed and are partially coded.

The code generation for SIM and SEQ statements has been completed. The algorithms for set storage allocation and usage have been revised to facilitate the usage of a set both as a mode bit pattern and as explicit numbers. The coding for the nondynamic cases of these revised schemes also has been completed. The algorithms are easily extendable for implementation of the dynamic costs.

Several straight forward but time-consuming routines of a general nature have been written. Among these are run-time arithmetic conversion routines for all combinations of conversions involving 32 and 64-bit floating point numbers, 48-bit signed integers, and both compile and run-time translations between set storage forms.

5.5.2 Glypnir

Features which allow routing and indexing have been implemented and debugged during this quarter. Several new debugging features have been added and a number of changes were made in the syntax to improve error detection and recover.

In addition, several changes were made in the semantics in an attempt to increase the speed and lower the core storage requirement, but these met with little success. B5500 mix time has been improved significantly, but the size and diversity of the compiler make it difficult to improve the computation time except by completely reorganizing the compiler.

The code which will implement subroutines has been completed but not debugged. The necessary changes which will allow the compiler to accept this code have been completed. However, due to a lack of PRT locations, a sizable delay may be necessary before this code can be inserted into the compiler.

The current compiler consists of approximately 8600 lines of ALGOL, has a core estimate of 15.8K words, and runs at approximately 350 cards per minute on the B5500.

5.5.3 Squash

Work has continued on the debugging aid for ALGOL, with expected completion in January 1969. An initial translator has been formed from the complete syntax and partial semantics. This is being debugged. The remaining semantic actions are being coded for incorporation into the translator early in January.

5.6 Scheduler and Simulators

5.6.1 Scheduler

The Scheduler for ILLIAC IV has been flow charted and a test version for the B5500 is being written. An up-to-date reference manual on the assembler was written and will be published by Burroughs. A special version of the assembler for use on the B5500-16PE ILLIAC IV at Paoli was written.

5.6.2 Simulators

During the past quarter, the new simulator was virtually completed. This simulator includes a simple loader that utilizes the loader pseudo-operations emitted by the assembler. Other features of the simulator are:

1. The simulator is half-load proof--the simulator may be restarted from a previous break-point if the execution is terminated.
2. Disk I/O facilities are included.
3. There will be a format and list facility to generate line printer output for debugging purposes.
4. There is an octal dump of all registers and memory.
5. A simulation scheduler has been written and demonstrated. The scheduler queues simulation requests and automatically initiates the simulator on a new job when it completes a previous job.

6. The simulator will multi-process well with other programs. A simulation, an ALGOL compile, and R/C (R/C is a multi-user text editor) have been successfully multi-processed.

The simulation ratio between B5500 execution time and simulated quadrant ILLIAC IV execution time is slightly less than 10^6, which is somewhat better than was expected on the 7094.

5.7 Control Data ALGOL

A project to consider using the Control Data 6600 as the control computer for ILLIAC IV has been initiated. A study of the conversion of Burroughs' Extended ALGOL to Control Data ALGOL is now being made. An immediate goal is to have the assembler and simulator running on the Control Data machine.

Control Data ALGOL is essentially ALGOL 60 with the ACM proposed ALGOL 60 input-output conventions. Burroughs' ALGOL is an extension of ALGOL 60 with such constructs as partial word operators, imbedded assignments, bit-by-bit logical operations, and much exotic input-output. It also has such gross syntactic structures as WHILE statements, DO...UNTIL statements, and CASE statements.

A preprocessor for Burroughs' ALGOL is being written. This preprocessor is processing the gross syntactic differences, the DEFINE and FILL statements, and many minor character set problems. A simple parser to generate subroutine calls for imbedded assignments and partial word operators will be added. Presently, work is being done on an interpretive method of simulating the Burroughs STREAM PROCEDURES. Eventually, the output of the preprocessor must be modified by hand to implement input-output calls and to change certain other constructs, which are not readily machine convertible, into CDC ALGOL.

5.8 CAT

5.8.1 Mesh Storage

For non-core-contained partial differential equation problems, the mesh of data points is stored on ILLIAC IV disk rather than in fast memory. Blocks of the mesh are brought into fast memory one at a time for processing. After the processing of a block (kernel calculations) is completed, processing of the next block in sequence is started. If the blocks do not span either dimension of the rectangular mesh, then two sweeping sequences are possible: sweeping by rows of blocks and sweeping by columns of blocks. In sweeping by rows, all of the blocks in a row i are read from disk in sequence. Then the blocks in row i + 1 are read in the same sequence. This process continues until all blocks in n rows are read. Sweeping by columns is similar.

Some problems may require successive sweeps of the mesh in alternating directions. Storing the blocks on disk in such a way that latency is minimized is a problem which is further complicated by the necessity for interblock communication. For the kernel to update a block, edges from the four neighboring blocks are required.

A scheme has been formulated which allows sweeping in any sequence of directions (by rows or by columns). Blocks are read from disk; kernel calculations are performed; and the updated blocks are written on disk. The scheme will accommodate a wide range of rectangular mesh sizes. Many particular schemes have been found over a sample of mesh sizes and with a latency ≤ 0.12 (12% of total problem time). The report of the investigation will be completed in January of 1969.

5.8.2 Subset of CAT

A model is being developed for optimizing code which is amenable to solution by linear programming. The model allows for reordering of statements from the original code and for many forms of non-obvious data manipulation to minimize the time the processor(s) waits for I/O.

This method is completely general in that it assumes no particular machine configuration. The necessary information, such as computation time for the various operations and secondary memory cycle time, are parameters. The number of (binary integer) variables resulting from any non-trivial code, unfortunately, is large ($\geq 10^3$). Present efforts are involved in attempting to reduce this number.

5.9 Mathematical Applications

5.9.1 Partial Differential Equations

In this quarter, the Tranquil version of an idealized particle-in-cell code for a plasma was completely debugged for syntax errors. The code is contained in ILLIAC IV Document No. 207 [5] which was written this quarter. Document 207 gives a description of an idealized particle-in-cell (PIC) code for a plasma.

The checkerboard storages scheme for the PIC codes was investigated at Los Alamos Scientific Laboratory. For 256 PE's using a 16x16 checkerboard on meshes of 20x100 or 30x100, the efficiency, due to distribution of particles, ranges from 33% to 43% for 30 time steps. If, instead of using 256 PE's, only 64 PE's are used, the efficiency ranges from 80% to 90%.

A preliminary investigation into alternate storage schemes for PIC type problems was completed this quarter. Written this quarter was another code and document concerning the modified successive overrelaxation (MSOR) method for solving both Laplace's and Poisson's difference equations [6]. The MSOR is much more efficient for small meshes on a parallel computer.

5.9.2 Generalized Ordinary Differential Equation Solver

The general aim of this work is to produce an ALGOL program which will be run on the B6500 and which will take some simple representation of a large system of ordinary differential equations as input and produce as output the ILLIAC IV machine code to perform the integration. Before generating the code, this program must decide the best method and storage allocation for the given problem. Since the program must also scan the input and do some compiling, one of the Translater Writer Systems on the B5500 is being used to build a scanner written in ALGOL. Work is proceeding on the algorithms necessary to perform this task and in studying the types of methods available. In the main, however, the methods will already exist as

skeleton codes which will be completed to form the desired machine language program.

5.9.3 Multidimensional Compressible Hydrodynamics

New methods for the solution of multidimensional, compressible, hydrodynamic flow problems are being investigated. In particular, methods having greater parallelism than those currently in use are desired. Much of this quarter was devoted to gaining experience with existing Eulerian methods and various velocity and density weighting schemes including OIL, donor, and Rich without the use of artificial viscosity. Also, preliminary results on a new continuous rezone technique have been obtained.

The above described techniques have been applied to test problems on the B5500. The test problems were selected from a report by Hicks of AFWL in which he describes several one-dimensional shock problems having analytic solutions [7]. The results obtained from "the shock tube problem" (V of the Hicks' report) are satisfactory. Greater difficulty has been encountered with the collision of two shocks (VI of the Hicks' report); although, the results obtained are comparable to those given by Lax-Wendroff.

5.9.4 Eigenvalues

5.9.4.1 Matrix Storage Methods

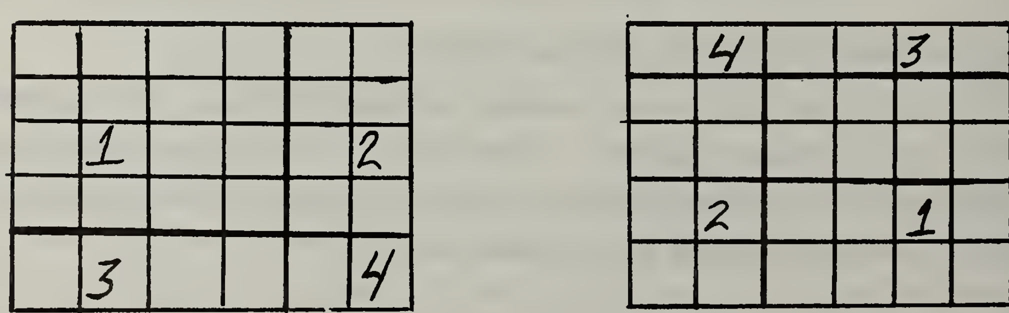
5.9.4.1.1 Matrix Storage for Jacobi's Method

A new method of matrix storage has been developed for Jacobi's method of finding eigenvalues for a matrix [8]. The storage method used is a variation of the triangular skew method. To illustrate this method, look at an 8x8 matrix stored in this manner:

$a_{1,1}$	$a_{1,2}$	$a_{1,3}$	$a_{1,4}$	$a_{1,5}$	$a_{1,6}$	$a_{1,7}$	$a_{1,8}$
$a_{2,8}$	$\boxed{\phantom{a_{0,0}}}$	$a_{2,2}$	$a_{2,3}$	$a_{2,4}$	$a_{2,5}$	$a_{2,6}$	$a_{2,7}$
$a_{3,7}$	$a_{3,8}$	$\boxed{\phantom{a_{0,0}}}$	$\boxed{a_{6,6}}$	$a_{3,3}$	$a_{3,4}$	$a_{3,5}$	$a_{3,6}$
$a_{4,6}$	$a_{4,7}$	$a_{4,8}$	$\boxed{\phantom{a_{0,0}}}$	$a_{6,7}$	$a_{7,7}$	$a_{4,4}$	$a_{4,5}$
$\boxed{\phantom{a_{0,0}}}$	$a_{5,5}$	$a_{5,6}$	$a_{5,7}$	$a_{5,8}$	$a_{6,8}$	$a_{7,8}$	$a_{8,8}$

In this storage method, each pair of successive diagonal elements is separated by one PE. The diagonal element $a_{\frac{n}{2}+1,\frac{n}{2}+1}$, in this case, $a_{5,5}$, is an exception in that it is separated from the previous diagonal element by 2 PE's. The elements beginning with $a_{\frac{n}{2}+2}$, $a_{\frac{n}{2}+2}$ form a wedge in the lower part of the matrix, as illustrated in the diagram above.

Assume that the origin of the matrix storage is the storage location of the element $a_{1,1}$ in the beginning. Call this location (0,0). After each shuffling of the second row and column, the origin is moved two locations to the right and one down. After one shuffling, the origin would be at (1,2) and after two shufflings, the origin would be at (2,4). The origin is again at (0,0) after $\frac{n}{2}$ shufflings. The effect of this on the updated matrix is to make the last two columns the first two columns of the new matrix, and make the last row the first row:



In this way, the elements of the updated matrix retain the same relationships as in the previous matrix.

Since the matrix has to be symmetric for Jacobi's method, this storage scheme allows all elements to be stored in an economical way. More precisely, for an n x n matrix, only $\frac{n}{2}+1$ rows of memory are required.

5.9.4.1.2 Matrix Storage for QR - Algorithm

Because the ordinary QR-algorithm is extremely inefficient for ILLIAC IV, a modification which does many origin shifts (instead of the customary two) in a single iteration, is being developed. The effect of the modification is to fill the matrix and allow a full Householder reduction at each step, making the method much more suitable for ILLIAC IV. Numerical experiments indicate that, although it is slower than ordinary QR by a factor of two, the accuracy of the eigenvalues from the two methods is equivalent.

5.9.4.2 Extended ALGOL Codes and Programs

Several codes and programs were written in Extended ALGOL. Codes have been written during this quarter, for Bessel function of first kind J_p (x) and Arctan (x). A program for finding the eignvalues and the eigenvectors for a symmetric matrix by using Jacobi's method was written in Extended ALGOL. Also written was a program to evaluate a matrix inversion by partitioning.

5.9.5 Polynomial Root Finding

During this quarter, an algorithm for determining whether a given square has a root in it was developed. There is a lemma in complex analysis which states:

$$S_n = \frac{1}{2\pi i} \int_c z^n \frac{f'(z)}{f(z)} d_z = \sum_{c=1}^{v} z^n ;$$

where z_i (i = 1,2, ..., v) are all the zeroes of f(z) which lie in the interior of the closed contour C. If N = 0, then S_o either has a value of zero or some positive integer which corresponds to the number of roots of the polynomial in the contour. S_o may then be approximated by the

trapezoidal rule. Thus, the algorithm consists of subdividing a given square region into 64 smaller squares, integrating on each of the squares, and repeating the process on those squares yielding a positive integration.

A Tranquil code for polynomial root finding is being debugged. When this is completed, a document will be written.

5.9.6 Special Functions Subroutine Library

Investigation was begun on writing subroutines in double precision or 128-bit mode. Meanwhile during this quarter, work continued on the development of a special functions subroutine library. A new 64-bit subroutine has been written for arctangent of a quotient of two numbers. The signs of the divisor and dividend determine in which quadrant the result will be. It was decided that the dividend should be in the A register and the divisor in the B register.

Also, codes have been written in the 32-bit mode for the exponential, sine, cosine, natural logarithm, arctangent, and square root functions. All codes previously mentioned in Quarterly Progress Reports have been written in ILLIAC IV assembly language. They have been successfully assembled and are waiting to be simulated. Documentation concerning all 32-bit and all 64-bit special function subroutines is being prepared and should be completed shortly.

5.9.7 Randon Number Generators

In this quarter, work began on the investigation of different randon number generators for ILLIAC IV. The most popular computing scheme to generate random numbers uses the congruential relation

$$x_{i+1} \equiv ax_i + c \qquad (\text{mod } m),$$

in which x_o is a positive integer, called the starting value; a is integer, called the multiplier; c is another integer; and m is a fourth integer, called the modulus, which is positive and greater than the other three in magnitude. When c = 0, the method is called multiplicative congruential; otherwise, it is called mixed congruential.

The advantages of the two methods will be discussed in a forthcoming document. The main concern of the document will center on the possible choices of the four parameters which will enable the random numbers to satisfy the statistical tests required of them.

5.9.8 Significant Digit Arithmetic

The work to implement significant digit arithmetic (SDA) on ILLIAC IV was begun this quarter. From the viewpoint of error analysis, the conventional normalized floating point arithmetic has several facets, and confusion often results from the failure to distinguish how many significant digits the evaluated result has. However, SDA operations, which are performed using unnormalized floating point arithmetic, are intended to facilitate the identification of significant digits in the result. In other words, the object of SDA is to keep track of as many significant digits throughout the whole arithmetic operation as there are given in the initial values.

During this quarter, some basic ideas for implementing Metropolis-type SDA on ILLIAC IV were proposed, and several versions for operating SDA have been tried. These results, including the mathematical basis and the assembly language codes for multiplication and division, are summarized in a document which will be printed soon.

5.9.9 Long Codes

In this quarter, the stability behavior of the autonomous system $\dot{x}(t) = Ax(t)$, where A is a constant matrix, was studied [9]. Also studied was the linear Hamiltonian system which is derived from the above general system. The algebraic condition on A is $A = J.S$ where $J = \left[\begin{array}{c|c} 0 & I \\ \hline -I & 0 \end{array}\right]$, and S is the time independent symmetric matrix $(H_{x_v x_{v+n}})$, $v = 1,2, \ldots, m$, where $m = \frac{n}{2}$.

During this quarter, an elementary algorithm was suggested for finding the eigenvalues of the constant matrix A. A fundamental matrix X

is to be built up from n-linearly independent observation vectors at a given time t, $x_1, x_2, \ldots, x_n$. The matrix A is formed by the multiplication $\dot{X}X^{-1}$. Once the matrix A is formed, the QR-algorithm would be used to determine the eigenvalues, and consequently, the stability behavior of the system.

Furthermore, the stability behavior of the general linear systems $\dot{x}(t) = A(t)\, x(t)$ was studied in detail. The elements of A(t) are assumed to be continuous in t and periodic with the same period, and the nature of the solutions of such systems described by the Floquet theory [10,11,12].

The stability criteria of the associated linear Hamiltonian systems,

$$\dot{x}_v = H_{x_{v+n}}, \quad \dot{x}_{v+n} = -H_{x_v} \qquad v = 1,2, \ldots, m$$

where $\dot{x}(t) = J.S(t)\; x(t)$, were investigated by using the novel approach of J. Moser [13]. In his paper, Moser showed that the eigenvalues of the monodromy matrix of a special fundamental matrix of the system

$$\dot{x}(t) = J.S(t)\; x(t)$$

constitute the invariants of the above system. A stability criterion could be easily established in the case of distinct eigenvalues; however, in the degenerate cases, one needs to refine the stability criterion. This may be done by establishing an additional set of invariants of the system, and the conditions of stability can be expressed in terms of the additional set of invariants.

5.10 Linear Programming

5.10.1 Introduction

Much of the work involved in the development of the LP system is of a continuing nature, and topics discussed in the last QPR have been subject to re-examination and revision from time to time. This condition will continue to occur. During this quarter considerable progress was made by the LP group. Formal definitions of the LPS algorithm and the

The advantages of the two methods will be discussed in a forthcoming document. The main concern of the document will center on the possible choices of the four parameters which will enable the random numbers to satisfy the statistical tests required of them.

5.9.8 Significant Digit Arithmetic

The work to implement significant digit arithmetic (SDA) on ILLIAC IV was begun this quarter. From the viewpoint of error analysis, the conventional normalized floating point arithmetic has several facets, and confusion often results from the failure to distinguish how many significant digits the evaluated result has. However, SDA operations, which are performed using unnormalized floating point arithmetic, are intended to facilitate the identification of significant digits in the result. In other words, the object of SDA is to keep track of as many significant digits throughout the whole arithmetic operation as there are given in the initial values.

During this quarter, some basic ideas for implementing Metropolis-type SDA on ILLIAC IV were proposed, and several versions for operating SDA have been tried. These results, including the mathematical basis and the assembly language codes for multiplication and division, are summarized in a document which will be printed soon.

5.9.9 Long Codes

In this quarter, the stability behavior of the autonomous system $\dot{x}(t) = Ax(t)$, where A is a constant matrix, was studied [9]. Also studied was the linear Hamiltonian system which is derived from the above general system. The algebraic condition on A is $A = J.S$ where $J = \left(\begin{array}{c|c} 0 & I \\ \hline -I & 0 \end{array}\right)$, and S is the time independent symmetric matrix $(H_{x_v x_{v+n}})$, $v = 1,2, \ldots, m$, where $m = \frac{n}{2}$.

During this quarter, an elementary algorithm was suggested for finding the eigenvalues of the constant matrix A. A fundamental matrix X

is to be built up from n-linearly independent observation vectors at a given time t, $x_1, x_2, \ldots, x_n$. The matrix A is formed by the multiplication $\dot{X}X^{-1}$. Once the matrix A is formed, the QR-algorithm would be used to determine the eigenvalues, and consequently, the stability behavior of the system.

Furthermore, the stability behavior of the general linear systems $\dot{x}(t) = A(t)\, x(t)$ was studied in detail. The elements of A(t) are assumed to be continuous in t and periodic with the same period, and the nature of the solutions of such systems described by the Floquet theory [10,11,12].

The stability criteria of the associated linear Hamiltonian systems,

$$\dot{x}_v = H_{x_{v+n}} \quad , \quad \dot{x}_{v+n} = -H_{x_v} \qquad v = 1,2, \ldots, m$$

where $\dot{x}(t) = J.S(t)\ \ x(t)$, were investigated by using the novel approach of J. Moser [13]. In his paper, Moser showed that the eigenvalues of the monodromy matrix of a special fundamental matrix of the system

$$\dot{x}(t) = J.S(t)\ \ x(t)$$

constitute the invariants of the above system. A stability criterion could be easily established in the case of distinct eigenvalues; however, in the degenerate cases, one needs to refine the stability criterion. This may be done by establishing an additional set of invariants of the system, and the conditions of stability can be expressed in terms of the additional set of invariants.

5.10 Linear Programming

5.10.1 Introduction

Much of the work involved in the development of the LP system is of a continuing nature, and topics discussed in the last QPR have been subject to re-examination and revision from time to time. This condition will continue to occur. During this quarter considerable progress was made by the LP group. Formal definitions of the LPS algorithm and the

associated data manipulations were almost completed. Some of the LP procedures were coded in ILLIAC IV assembly language and other coding is in progress. The progress to date is summarized below.

5.10.2 Data Input

The initial steps required to define and develop a formal language for use in matrix generation were taken. The requirements for a meaningful matrix-generation system were examined, and there will be a major effort in this area in the near future.

5.10.3 Preprocessing of Data

The procedures necessary to pack and store input data have been defined. The non-zero coefficients will be assigned to specific PE's and will be tagged accordingly for subsequent disk storage. The preprocessing will be done on the B6500.

5.10.4 Storage

The data storage procedure has been well defined and is as follows. Since individual columns of the coefficient (A) matrix are needed for d_j (reduced price) computation, for updating with previously computed n (product form inverse) vectors, and for subsequent entry into the basis, each column should be stored so that its non-zero coefficients are distributed across PE's. This is attained by first assigning rows to adjacent PE's, starting with the most dense, in such a manner as to evenly distribute the non-zero elements of A between PE's. This is subject to the constraint that elements of a given row can be assigned to only one PE. The coefficients of the rows assigned to each PE are then sorted by columns within the given PE. This storage scheme assures that operations done on columns will be done in parallel, since the elements of a given column will be stored in several PE's.

5.10.5 The LPS Algorithm

Progress in connection with the algorithm has been significant. The procedures for d_j computation, multiple pricing, pivot selection,

and vector updates using the product of previously generated n-vectors have been coded in ILLIAC IV assembly language. The mathematical procedures involved in right-hand side ranging, parametric programming, and bounding of variables have been examined and flow charted. An intensive examination of inversion techniques for use in the needed REINVERT subroutine has been initiated, and it is continuing. The subject of tolerances on computed values has been approached, and will be more carefully examined.

5.11 Radar Processing Applications

Efforts during this period have been devoted to the Kalman Filter Code debugging, to understanding a new, simpler version of Kalman Filtering developed by Lincoln Labs, and to converting the code for the designation mode to 32-bit floating point operation. Work began on programming Fast Fourier Transform in 32-bit floating point for ILLIAC IV to be used in the discrimination mode and on investigating the problems in radar scheduling and the effects on using ILLIAC IV for the scheduling.

Progress in the debugging of the Kalman Filtering program in ILLIAC IV assembly language has been slow. There is presently no assembler simulator that can handle the 32-bit floating arithmetic which is used in the Kalman programs. The program has been assembled on a comparable assembler and is essentially ready for the execution simulator. To facilitate faster and easier debugging of the assembly language code, the Kalman Filer Program has been written in ALGOL to run on the B5500 computer. This program will supply data for checking out each section of the assembly language program. The ALGOL program is running, but it is not completely debugged. With the aid of data supplied by Lincoln Labs, this program should be working by early February. At that time, it is probable that the ILLIAC IV assembler and simulator for 32-bit mode arithmetic will be in operation. The ILLIAC IV assembly language of the Kalman Filter can then be debugged by using the simulator and data generated for the ALGOL version.

The Tranquil version of the Kalman Filter written for ILLIAC IV will not be debugged until the compiler is completed to the point that it can handle all the Tranquil statements used in the program. This will not

be accomplished for sometime; however, when complete, it will be a good test case for using higher level language as opposed to assembly language on ILLIAC IV for radar processing.

A new version of the Kalman Filter which uses only approximately 1/2 to 2/3 the time of the old version has been developed by Lincoln Labs. It does everything in spherical coordinates instead of the combination of spherical and cortesian coordinates. This new version will be programmed in ILLIAC IV assembly language using 32-bit floating point arithmetic.

The designation mode which was in fixed point arithmetic (described in ILLIAC IV Document 173 [14]) is being reprogrammed by the new graduate student into 32-bit floating point arithmetic. This mode uses a .1 second and a 1 second filtering technique of a cloud in respect to cloud center to try to filter objects from shaft.

The Fast Fourier Transform (FFT) is being programmed for ILLIAC IV as a subroutine using 32-bit floating point arithmetic. It will be set up in the following two forms: 1. returns from the same object are spread across PE's, and the different object on which data is obtained goes down memory of each PE; and 2. the objects are spread across PE's, and data on each goes down memory. Many different forms of discrimination make use of FFT, and it will be needed for ILLIAC IV.

The radar scheduling problem and the efficiency of doing this portion of the radar processing task on ILLIAC IV is being investigated. Several portions of this problem seem to be sequential; therefore, it may be more efficient to handle a large part of the task on a standard computer which is controlling ILLIAC IV. However, the interplay of the two computers on this problem is being studied.

5.12 Seismic Signal Processing

During this quarter, a study of the programming requirements necessary to compile a seismic signal processing package was undertaken. This resulted in defining the specifications for approximately twenty-five programs. Several of these programs, such as those involving matrix manipulations, may be able to incorporate current programs from other study areas.

This investigation has indicated that to make the seismic programs most functional, they must be written so that they can be used in areas other than seismic processing with a minimum amount of modification. Some of the specialized seismic programs will have applications in other areas--such as programs for convolution, filter design, deconvolution, power spectrum analysis, and correlation analysis.

5.13 Weather

The finite difference portions of the General Circulation Model Benchmark provided by the National Center for Atmospheric Research have been coded in a combination of Tranquil and assembly language. These arithmetic subroutines must be connected by a machine language control program which will provide necessary parameters and which will make adjustments for unusual grid spacing caused by mountain blocking and stability criterion near the poles.

5.14 Graphics

5.14.1 Introduction

The graphic activity is divided into software and hardware sections. The software activity is concerned with developing algorithms and techniques to apply ILLIAC IV to graphic problems. A specific task is presented in the following software section. The hardware activity is concerned with the peripheral equipment for the ILLIAC IV computer. This equipment will be used to generate output which will be used by the programmer for debugging his program and providing graphic output in the form of contour maps and various graphs. Consideration is also being given to providing a graphic terminal to provide program status to the operators.

5.14.2 Software

A problem of particular, current interest in the area of computer graphics is the so-called hidden line problem. This problem is concerned with computer determination of the visible and invisible parts of three dimensional objects when viewed from an arbitrary point.

A simple, fast algorithm is essential because there are many applications in which it is desired to view a moving object of a CRT. However, the problem has proved to be unusually complex. The following two structures are mainly concerned:

1. three dimensional polyhedra consisting of irregular and opaque polygons, where only line segments with end-point pairs, one pair for each line, are assumed as the input data because of the construction of three dimensional objects from line drawing and

2. three dimensional objects including curved surfaces decomposed into a lot of triangles, where each decomposed triangle of the curved surfaces must be completely specified as the input data.

The algorithm for polygon generation is completely specified, and the list structure format is used for data structure of pictures.

To motivate the description of the algorithm of the hidden line problem, the factors which determine whether or not which parts of objects are visible can be considered. In the case of a single polyhedron, call it the <u>local hidden line problem</u>. For a compound structure of polyhedra, locally visible polygons of a polyhedron may become invisible or partially invisible by the polygons of another polyhedron. Call this the <u>global hidden line</u>. In either case, if the relationship between two polygons is determined, the hidden line problem may be solved. The relationship between two polygons is defined as follows:

(a) One is enclosing the other;

(b) One is involved by the other; or

(c) There is no intersection between them.

The program is being implemented on B5500. Additional documentation will be provided by a masters thesis being written in this area. The thesis should be available next quarter.

5.15 Statistical Packages

During this quarter, design was begun for a statistical system for ILLIAC IV. The orientation of the system will be toward techniques which are applicable to large-scale data bases and which are used repeatedly. This is in apposition to a multiplicity of small techniques which are oriented toward editing data or output or which are convenience programs with little computational content. It is intended that the computation aspects of the statistical system will rely heavily upon matrix operations generated by the matrix algebra group, when feasible.

The following functions have been chosen as a basic set:

(a) Data transformation,

(b) General Matrix operations,

(c) Correlation,

(d) Multiple Regression,

(e) Analysis of Variance
(by itself regarded currently as a very significant problem),

(f) Principal Axis Factor Analysis,

(g) Autocorrelations, and

(h) Frequency counting and Distribution Analysis.

It is expected that a limited number of functions may be added to this list in the future.

Under consideration, also are tools or languages for dealing with large scale data structures more complicated than the rectangular arrays customarily used for input to the above techniques. This system is still in the speculative stage.

5.16 ILLIAC IV Education

5.16.1 Introduction

The ILLIAC IV education for this quarter consisted of organizing and presenting two courses of instruction--one covering B5500 and

ALGOL material and the other covering material concerned with the ILLIAC IV machine, its languages, and its applications. In addition, the education group has been updating educational material for the use of the ILLIAC IV staff and persons interested in the ILLIAC IV. This quarter also saw an attempt to organize all programs written or being written for ILLIAC IV into a library, which is to be available to anyone interested.

5.16.2 B5500 and ALGOL

This course was offered to acquaint ILLIAC IV personnel with the B5500 and ALGOL. The topics covered in this course were: the basic elements of ALGOL and special B5500 Extended ALGOL features, descriptions and instructions for use of the peripheral units, a detailed description of I/O, and a discussion of the master control program of the B5500.

5.16.3 CS 491-D

This course was a graduate seminar course CS 491-D, which was offered to ILLIAC IV personnel. (It will be offered University-wide for the spring semester.) This course included a discussion of the hardware of ILLIAC IV, coverage of its assembly language and Tranquil, a discussion of the Assembler, and extensive coverage of ILLIAC IV applications.

REFERENCES

[1] Beals, Alan J. "The Generation of a Deterministic Parsing Algorithm." Master's Thesis, Report No. 304. Urbana, Illinois: Department of Computer Science, University of Illinois at Urbana-Champaign, 1969.

[2] Budnik, Paul P. "TRANQUIL Arithmetic." Master's Thesis, Report No. 296. Urbana, Illinois: Department of Computer Science, University of Illinois at Urbana-Champaign, 1969.

[3] Muraoka, Yoichi. "Storage Allocation Algorithms in the TRANQUIL Compiler." Master's Thesis, Report No. 297. Urbana, Illinois: Department of Computer Science, University of Illinois at Urbana-Champaign, 1969.

[4] Wilhelmson, Robert B. "Control Statement Syntax and Semantics of a Language for Parallel Processors." Master's Thesis, Report No. 298. Urbana, Illinois: Department of Computer Science, University of Illinois, 1969.

[5] Rudsinski, L. E. "A Tranquil Version of an Idealized Particle-In-Cell Code for a Plasma." ILLIAC IV Document Number 207, (November 25, 1968).

[6] Rudsinski, L. E. "Tranquil Code for the Modified Successive Overrelaxation (MSOR) Method for Solving Laplace's Difference Equation." ILLIAC IV Document Number 208, (December 24, 1968).

[7] Hicks, Darrell. "Hydrocode Test Problems." AFWL-TR-67-127. New Mexico: Kirtland Air Force Base.

[8] Kuck, David. "ILLIAC IV Organization, Programming, and Storage Allocation." Lecture given before the Mathematics, Computation, and Computers Conference, University of Illinois at Urbana-Champaign, Urbana, Illinois, November, 1968.

[9] Malkin, I. G. "Theory of Stability of Motion." Translation. Moscow and Leningrad: State Publishing House of Technical-Theoretical Literature, 1952.

[10] Coddington, E. A., and Levinson, N. Theory of Ordinary Differential Equations. New York: McGraw Hill, 1955.

[11] Wintner, A. The Analytical Foundations of Celestial Mechancis. Princeton: Princeton University Press, 1941.

[12] Lefschetz, S. Differential Equations, Geometric Theory. New York: International Science Publishers, Incorp., 1957.

[13] Moser, J. "New Aspects in the Theory of Stability of Hamiltonian Systems," Communications on Pure and Applied Mathematics. XI, (1958), 81-114.

[14] Knapp, M. A., Ackins, G.N., and Thomas, John. "Application of ILLIAC IV to Urban Defense Radar Problem." ILLIAC IV Document Number 173, (February 21, 1968).

6. NUMERICAL METHODS, COMPUTER ARITHMETIC AND ARTIFICIAL LANGUAGES (Supported in part by the National Science Foundation under Grant No. NSF-GJ-217.)

6.1 Computerized Mathematics

Except for a few final tests, the assembly language theorem-proving program was completed this quarter. We have written the program so as to require only minor modifications to perform hyper-resolution or resolution with merging, two modified procedures based on resolution. We plan to implement these during the next quarter.

We have been making corrections on our EØL program. We hope to be able to finish it during this quarter and then collaborate with J. Nievergelt of this department, who is in charge of the EØL project, to test the efficiency and find the weak areas of the EØL language by observing the operation of the two theorem-proving programs.

Along theoretical lines, using the methods given in my masters thesis, we have been considering ways of reducing proof trees to a canonical form. The simplest form would be a vine, i.e., from each node there emanates a branch with only one clause. Let (A,B) indicate a resolvent of clause A with clause B. Then a vine can be written (((...((A,B),C), ...), X), Y) where A, B, C, ... X, Y are clauses. By reducing proof trees to a special form, it is hoped the search for a proof will be simplified. We know this reduction is not always possible because of a problem called merging. Suppose the clauses A:p,q and B:p,~q,r were resolved. The result is C:p,r and the two literals p merged. If we now resolve C with D:~p,s, the result is r,s. If D was the root of a vine, the next step would be to rewrite the deduction

((A,B),D) as either ((A,D),B) or ((D,B),A) to obtain the end result r,s. However, because of the merge, the only way to rewrite that deduction to obtain p,s is ((A,D),(B,D)), which is still not of vine form. Thus, if the merge literal is used to resolve again, the vine structure cannot be obtained. We will continue our reseach in this area in the next quarter.

(L. J. Henschen)

6.2 Numerical Solution of Singular Integral Equations

During the quarter work was initiated on the consideration of techniques for the numerical solution of singular integral equations.

The results on convergence and error bounds for approximate solutions of integral equations by Anselone and Moore ("Approximate Solutions of Integral and Operator Equations," J. of Math. Anal. and Appl., Vol. 9, (1964), pp 268-277) were studied together with other associated papers by Anselone in this area. In a later paper (P. M. Anselone, "Uniform Approximation Theory for Integral Equations with Discontinuous Kernels," SIAM J. Numer. Anal., 4 2(1967), pp 245-253) he extends these results from continuous kernels to kernels which are discontinuous but still Riemann-integrable.

Atkinson ("Numerical Solution of Fredhelm Integral Equations of the Second Kind," SIAM J. Num. Anal., 4, 3(1967), pp 337-348) extends the general result still further to weakly singular kernels using generalized quadrature formulas based on product integration.

However, none of these papers appear to offer an immediate generalization to the case of a strongly singular kernel (Cauchy kernel) and work is now be beginning on a study of Muskhelishvili ("Singular Integral Equations,"" P. Noordhoff N. V., Groningen, Holland (1958)) in order to determine whether techniques for the analytical solution of singular integral equations might be exploited to obtain a uniform approximation theory for strongly singular kernels, as was suggested by Anselone in a recent lecture.

(A. W. McInnes)

6.3 Study of Methods of Selection of Quotient Digits During Digital Division

This study concerns a class of division technique in which redundancy is introduced into the representation of the quotient thus permitting quotient digits to be selected by inspection of estimates of the full-precision divisor and partial remainder. For example, a radix 256 quotient digit (8 bits) may be generated knowing only the first 11 bits of the divisor and the first 12 bits of the partial remainder.

A major goal of this work is to develop quotient selection mechanisms, and then to perform cost-effectiveness studies of the most promising. Two major classes of techniques have been proposed: (1) direct table look-up, and (2) table look-up of the inverse of the estimate of the divisor followed by a multiplication of the inverse and the estimate of the partial remainder.

Research during this quarter centered upon the first class of tecniques -- direct table look-up. An algorithm was developed and coded in FORTRAN IV which will generate an estimate of the logic requirements. The basic factors which determine the logic requirements and which serve as parameters to the program are as follows:

1) The radix

2) The range of the divisor

3) The direction of the truncation error in the estimate of the divisor and partial remainder, i.e., strictly positive, strictly negative, or either positive or negative

4) The amount of redundancy permitted in the quotient. The allowable quotient digits belong to the set $\{0, \pm 1, \pm 2, \ldots, \pm n\}$. A measure of redundancy is the ration $n/(r-1)$ where r is the radix.

The program (QUOtient SELection -03) has been completed and run for several variations of parameters. Production runs with other combinations of parameter values are continuing. To date it has been found that for a divisor (d) range given by $1/2 \leq d < 1$, an error in the estimate of the partial remainder of $\pm 1/16$, and a redundancy ratio $n/(r-1) = 2/3$, the cost of logic is approximately proportional to $2.7r^2$.

(D. E. Atkins)

6.4 Automatic Function Generation*

In arithmetic processors the cost and size reduction and increasing reliability of integrated circuit logic elements permits the replacement of programmed subroutines by built-in function generators. This research is a study of the efficient generation of elementary functions frequently required in a scientific binary computer. These functions include division, square rooting, exponentiation, computation of logarithms, and evaluation of elementary circular functions. Only the fractional part of a floating point operand is being considered. The basic technique is to reduce function evaluation to a sequence of comparisons, additions, and shifts. If a shift requires much less time than an addition, then the speed is a direct function of the number of additions required. It is proposed to reduce the number of additions required, averaged over all allowed operands, to a (near) minimum.

---

* The research on this topic is reported here because of its pertinance to the goals of NSF Grant GJ-217. Mr. DeLugish is a candidate for the degree of Doctor of Philosophy in Electrical Engineering, with support from the Department of Electrical Engineering. It is anticipated that some computer use will shortly become necessary, to be supported by NSF Grant GJ-217.

Algorithms for division, exponentiation, and computation of natural logarithms have been devised. Although an analysis to determine the actual efficiency of these algorithms has yet to be attempted, it appears that each can be completed in slightly more than $R/3$ addition cycle times, where R is the length of the fractional part of the operand. These algorithms require storage of a small number (approximately R) of precomputed constants.

Research is continuing on an algorithm for square rooting.

(B. G. DeLugish)

7. SWITCHING THEORY AND LOGICAL DESIGN

This quarter was spent in preparing the following reports of our work:

1. Logical design of an optimum network by integer linear programming - Part II, by S. Muroga and T. Ibaraki

2. An implicit enumeration program for zero-one integer programming, by T. Ibaraki, T. K. Liu, and C. R. Baugh and S. Muroga

3. An optimum network design using NOR and NOR-AND gates by integer programming, by C. R. Baugh, T. Ibaraki, T. K. Liu and S. Muroga

4. A code for zero-one integer linear programming by implicit enumeration, by T. K. Liu.

A computer program which can be used for general (0,1) - variable integer programming program has been prepared, based on the implicit enumeration method with modification and introduction of new ideas. The program was initiated for logical design but it can be used for any other (0,1) - variable integer programming problems. The program is named ILLIP (Illinois Integer Programming.) The basic ideas are discussed in the above report 2 and the programming manual is available as the above report 4 which is Liu's Master degree thesis.

S. Muroga

8. SOUPAC SECTION
(Statistically Oriented Users Programmers and Consultants)

During the time period from October 1 through December 31, 1968, there were 710 consultations arranged with members of the SOUPAC Section, and this averages to be more than eleven appointments per day. Most of these discussions were concerned with the utilization of the IBM 7094 SSUPAC collection of programs, and a counter in SSUPAC indicated more than 7500 references during the time period. In addition more than 100 manuals of instruction for SSUPAC were distributed.

Despite this continuing interest in SSUPAC, much of the effort of the Statistical Consultants was devoted to writing programs for the IBM 360 SOUPAC collection of statistical and data manipulative procedures. During December, seven of the individual programs of SOUPAC were announced as operational: Autocorrelations, Biserial Correlations, Zero-level Correlations, Multiple Correlations, Matrix Operations, Multiple Discriminant Analysis, and Varimax. Of course adequate documentation for the programs does not exist. Nevertheless, two brief descriptions, entitled "Description and Organization of SOUPAC" and "Introduction to SOUPAC", were reproduced and made available for distribution.

Kern W. Dickman
Project Director

9. IBM SYSTEM/360 SERVICE

(Supported in Part by the National Science Foundation under Grant No. NSF-GP-7634.)

9.1 Log Summaries

Table I - IBM 360/20
Summary of Use
October, 1968

Unscheduled Engineering		38:56
Scheduled Engineering		4:05
Maintenance		2:47
Production		272:53
Operator Training		:56
Idle		47:51
Air-Conditioning		:47
	Total	368:15

Table II - IBM 360/20
Summary of Machine Errors
October, 1968

2560 MFCM		21
	Total	21

Table I - IBM 360/50-75

Summary of Use

October, 1968

Unscheduled Engineering	35:00		
Scheduled Engineering	53:31:07		
Maintenance	1:51:22		
Idle	3:08:00		
		Subtotal	59:05:29
Total Use			
Training and Education		130:46:40	
System Improvement and Modifications		99:05:48	
System Updating		54:04:58	
University Administrative Overhead Use		1:11:10	
Overhead		17:49:15	
Refund		39:20	
Customer Use			
In System	137:43:41		
Special Short Shots	1:44:06		
Customer Use Total		139:27:47	
		Total Use	443:04:58
		Total Time	502:10:27

Table II - IBM 360/50-75

Summary of Errors

October, 1968

2540 Card Reader Punch		1
	Total	1

IBM 360/75 TABLE III OCT., 1968

DEPT[1]	NUMBER OF RUNS			NUMBER OF SPECS			IBM 360/75 USAGE IN HH:MM:SS		
	T AND E[2]	RES	TOTAL	T AND E[2]	RES	TOTAL	T AND E[2]	RES	TOTAL
AAE	670	60	730	5	2	7	7:21:30	2:50:26	10:11:56
ACCY	321	17	338	1	3	4	1:17:38	0:18:27	1:36: 5
AGE	328	56	384	3	2	5	5:58:24	0:33: 5	6:31:29
AGEC	0	224	224	0	7	7	0: 0: 0	3:40:59	3:40:59
AGRON	449	246	695	1	7	8	1:33:44	1:21:21	2:55: 5
ANS	0	44	44	0	2	2	0: 0: 0	0:15:28	0:15:28
ARCH	157	0	157	2	[illegible]	2	0:49:19	0: 0: 0	0:49:19
ASTR	0	121	121	0	5	5	0: 0: 0	1: 4:37	1: 4:37
BECBSR	0	9	9	0	1	1	0: 0: 0	0: 4: 8	0: 4: 8
CE	979	747	1726	11	36	47	8:30: 7	18:41:52	27:11:59
CHE	81	715	796	5	24	29	0:21:49	15:56:22	16:18:11
CRC	0	29	29	0	1	1	0: 0: 0	0:22:25	0:22:25
DCS	16206	217	16423	15	10	25	74:38:22	6:25: 3	81: 3:25
DS	0	22	22	0	1	1	0: 0: 0	0:13:19	0:13:19
DUE	32	0	32	1	0	1	0: 5:19	0: 0: 0	0: 5:19
ED	0	81	81	0	2	2	0: 0: 0	1: 2: 1	1: 2: 1
EDADM	0	35	35	0	1	1	0: 0: 0	0:54:40	0:54:40
EDPSY	48	0	48	2	0	2	0:13: 4	0: 0: 0	0:13: 4
EE	1653	789	2442	14	33	47	12:56:15	9:16:11	22:12:26
ENGADM	0	90	90	0	2	2	0: 0: 0	1: 1:47	1: 1:47
ENGCST	0	19	19	0	1	1	0: 0: 0	0:11:35	0:11:35
ENGH	311	0	311	1	0	1	1: 6:54	0: 0: 0	1: 6:54
ENGRDV	0	6	6	0	1	1	0: 0: 0	0: 1:42	0: 1:42
FT	0	12	12	0	2	2	0: 0: 0	0: 5:54	0: 5:54
GENE	164	0	164	4	0	4	0:46:39	0: 0: 0	0:46:39
GEOG	0	17	17	0	2	2	0: 0: 0	0:20:31	0:20:31
GEOL	42	22	64	3	2	5	0:20:43	0:14:41	0:35:24
GER	0	0	0	0	1	1	0: 0: 0	0: 7:56	0: 7:56
GSBA	377	0	377	3	0	3	2:39:13	0: 0: 0	2:39:13
HORT	0	14	14	0	1	1	0: 0: 0	0:16:56	0:16:56
ICR	0	72	72	0	1	1	0: 0: 0	0:54:35	0:54:35
IE	7	0	7	1	0	1	0: 4:59	0: 0: 0	0: 4:59
ILLCF	0	2	2	0	1	1	0: 0: 0	0: 0:37	0: 0:37
ILLDMH	0	289	289	0	1	1	0: 0: 0	10:10:30	10:10:30
INADM	0	2	2	0	1	1	0: 0: 0	0: 0:14	0: 0:14
MATH	0	7	7	0	2	2	0: 0: 0	0: 6:12	0: 6:12
MATRL	0	701	701	0	23	23	0: 0: 0	11:38:26	11:38:26
MCBIO	0	16	16	0	1	1	0: 0: 0	0: 7:26	0: 7:26
ME	2566	183	2749	15	9	24	10:54: 5	2: 5:38	12:59:43
MMPE	0	76	76	0	2	2	0: 0: 0	1:44:33	1:44:33
NHS	0	60	60	0	2	2	0: 0: 0	1:41:41	1:41:41
NUCE	105	140	245	3	6	9	0:33:55	1:46:32	2:2[illegible]:27

CIR	0	224	224		1	1	0: 0: 0	2:10: 1	2:10: 1
CRME	0	2	2	0	1	1	0: 0: 0	0: 1: 5	0: 1: 5
PHYB	0	60	60	0	1	1	0: 0: 0	1:21:11	1:21:11
PHYCS	9	609	618	1	14	15	0: 8:54	7:13:21	7:22:15
PHYX	0	1055	1055	0	6	6	0: 0: 0	22:37:13	22:37:13
FLPA	0	13	13	0	1	1	0: 0: 0	0: 2: 4	0: 2: 4
POLS	12	19	31	1	1	2	0:11:35	0: 2:10	0:13:45
PSYCH	2	4	6	1	2	3	0: 0:37	0: 0:58	0: 1:35
REC	0	5	5	0	1	1	0: 0: 0	1:28:20	1:28:20
SCONS	0	70	70	0	1	1	0: 0: 0	0:52:23	0:52:23
SGS	0	240	240	0	2	2	0: 0: 0	1: 9: 3	1: 9: 3
SOC	0	3	3	0	1	1	0: 0: 0	0: 1:19	0: 1:19
SRL	0	111	111	0	3	3	0: 0: 0	1:21: 6	1:21: 6
SWS	0	65	65	0	5	5	0: 0: 0	1:51:54	1:51:54
TAM	5	126	131	1	8	9	0: 1:47	1:15:34	1:17:21
VPP	0	1	1	0	1	1	0: 0: 0	0: 0: 7	0: 0: 7
WPGU	0	2	2	0	1	1	0: 0: 0	0: 1:16	0: 1:16
ZOOL	63	90	153	1	5	6	0:11:48	0:52:38	1: 4:26
SUBTOTAL	24587	7839	32426	95	253	348	130:46:40	137:43:41	268:30:21
DCSSYS[3]	0	3559	3559	0	21	21	0: 0: 0	99: 5:48	99: 5:48
OVRHED[4]	0	6788	6788	0	1	1	0: 0: 0	17:49:15	17:49:15
REFUND[5]	0	11	11	0	1	1	0: 0: 0	0:39:20	0:39:20
SSUAD[6]	0	52	52	0	1	1	0: 0: 0	1:11:10	1:11:10
XDCS[7]	0	291	291	0	3	3	0: 0: 0	54: 4:58	54: 4:58
XMAINT[8]	0	14	14	0	4	4	0: 0: 0	59: 5:29	59: 5:29
XSSS[9]	0	278	278	0	1	1	0: 0: 0	1:44: 6	1:44: 6
TOTAL	24587	18832	43419	95	285	380	130:46:40	371:23:47	502:10:27

1 See list of departmental codes following
2 Training and Education
3 System Improvement and Modifications
4 System Use in Excess of User Estimated Usage
5 Rerun of User Jobs Due to Machine or System Error
6 University Administrative Overhead Use
7 System Updating
8 System Maintenance
9 Special Short Shots

Table I - IBM 360/20

Summary of Use

November, 1968

Unscheduled Engineering		10:35
Scheduled Engineering		11:08
Maintenance		4:22
Production		260:36
Operator Training		11:59
Idle		174:20
	Total	473:00

Table II - IBM 360/20

Summary of Machine Errors

November, 1968

2560 MFCM		12
2203 Printer		1
Faulty paper		1
	Total	14

Table I - IBM 360/50-75

Summary of Use

November, 1968

Unscheduled Engineering	9:30:00		
Scheduled Engineering	26:24:06		
Maintenance	4:28:13		
Air Conditioning	2:05:00		
Idle	20:00		
		Subtotal	42:47:19
Total Use			
Training and Education		121:00:49	
System Improvement and Modifications		91:37:23	
System Updating		52:56:31	
University Administrative Overhead Use		1:18:56	
Overhead		15:19:00	
Refund		29:52	
Customer Use			
In System	127:18:03		
Special Short Shots	1:52:10		
Customer Use Total		129:10:13	
		Total Use	411:52:44
		Total Time	454:40:03

Table II - IBM 360/50-75

Summary of Errors

November, 1968

2540 Card Reader Punch		1
50 CPU		2
1050 Console Typewriter		1
	Total	4

IBM 36 /75 TABLE III NOV., 1968

DEPT[1]	NUMBER OF RUNS			NUMBER OF SPECS			IBM 360/75 USAGE IN HH:MM:SS		
	T AND E[2]	RES	TOTAL	T AND E[2]	RES	TOTAL	T AND E[2]	RES	TOTAL
AAE	575	173	748	5	4	9	6:24:41	3:52:20	10:17: 1
ACCY	205	6	211	1	1	2	0:45:33	0: 5:33	0:51: 6
AGE	144	69	213	6	3	9	3:43: 0	0:45:24	4:28:24
AGEC	0	263	263	0	11	11	0: 0: 0	6:14: 3	6:14: 3
AGRON	500	148	648	1	7	8	2:22:47	0:57:10	3:19:57
ANS	0	78	78	0	2	2	0: 0: 0	0:53:48	0:53:48
ARCH	196	1	197	3	1	4	1:12: 4	0: 0:52	1:12:56
ASTR	0	65	65	0	5	5	0: 0: 0	0:41:51	0:41:51
BECBSR	0	33	33	0	1	1	0: 0: 0	0:29:29	0:29:29
CE	1429	896	2325	14	42	56	9:26: 8	19: 8:38	28:34:46
CHE	544	812	1356	5	27	32	3:21:59	15:28:55	18:5 :54
CRC	0	42	42	0	1	1	0: 0: 0	0:19:23	0:19:23
CSL	0	16	16	0	2	2	0: 0: 0	0: 6:23	0: 6:23
DCS	12475	284	12759	16	9	25	55:46:52	11:41:43	67:28:35
DS	0	6	6	0	1	1	0: 0: 0	0: 7:29	0: 7:29
DUE	2	0	2	1	0	1	0: 0:19	0: 0: 0	0: 0:19
ECON	0	5	5	0	2	2	0: 0: 0	0:31:47	0:31:47
ED	0	16	16	0	3	3	0: 0: 0	0:11:49	0:11:49
EDPSY	72	0	72	2	0	2	0:26:38	0: 0: 0	0:26:38
EE	2809	565	3374	14	28	42	17:47:18	5:14:19	23: 1:37
ENGADM	0	56	56		2	2	0: 0: 0	0:45: 7	0:45: 7
ENGCST	0	55	55	0	1	1	0: 0: 0	0:59:24	0:59:24
ENGH	291	0	291	1	0	1	0:35:57	0: 0: 0	0:35:57
ENGRDV	0	6	6	0	1	1	0: 0: 0	0: 2:34	0: 2:34
FT	0	23	23	0	3	3	0: 0: 0	0: 8:49	0: 8:49
GENE	456	0	456	4	0	4	2:46:46	0: 0: 0	2:46:46
GEOG	0	49	49	0	2	2	0: 0: 0	0:21:27	0:21:27
GEOL	160	22	182	3	1	4	1: 8:35	0: 8: 9	1:16:44
GER	0	4	4	0	1	1	0: 0: 0	0: 2:30	0: 2:30
GSBA	290	0	290	3	0	3	2: 0:55	0: 0: 0	2: 0:55
HORT	0	1	1	0	1	1	0: 0: 0	0: 3:24	0: 3:24
ICR	0	92	92	0	2	2	0: 0: 0	1: 0: 3	1: 0: 3
IE	48	0	48	1	0	1	0:16:15	0: 0: 0	0:16:15
ILLCF	0	3	3	0	1	1	0: 0: 0	0: 1:49	0: 1:49
ILLDMH	0	222	222	0	1	1	0: 0: 0	4:36:26	4:36:26
INADM	0	18	18	0	1	1	0: 0: 0	0: 7: 6	0: 7: 6

MATH	0	28	28		1	1	0: 0: 0	0:10:23	0:10:23
MATRL	0	574	574		22	22	0: 0: 0	10:23: 7	10:23: 7
MCBIO	0	10	1		1	1	0: 0: 0	0: 3:48	0: 3:48
ME	1790	250	2040	19	9	28	9:16:19	2:25: 2	11:41:21
MKTG	0	10	10		1	1	0: 0: 0	0: 2:22	0: 2:22
MMPE	0	53	53	0	2	2	0: 0: 0	2:56:54	2:56:54
NHS	0	11	11	0	1	1	0: 0: 0	0:17:24	0:17:24
NUCE	27	197	224	3	9	12	0:10: 4	1:32:39	1:42:43
OIR	0	77	77	0	1	1	0: 0: 0	0:40:55	0:40:55
ORME	0	10	10		1	1	0: 0: 0	0: 3: 3	0: 3: 3
PHYB	0	87	87	0	2	2	0: 0: 0	2:18:21	2:18:21
PHYCS	106	603	709	3	14	17	0:16:44	10:22: 3	10:38:47
PHYX	0	916	916		27	27	0: 0: 0	17:30: 9	17:30: 9
PLPA	0	26	26		1	1	0: 0: 0	0: 4: 1	0: 4: 1
POLS	20	14	34	1	1	2	0:30: 1	0: 6:53	0:36:54
PSYCH	0	53	53	0	4	4	0: 0: 0	0:29:19	0:29:19
REC	0	20	20		2	2	0: 0: 0	0:26:18	0:26:18
SCONS	0	27	27	0	1	1	0: 0: 0	0:33:26	0:33:26
SGS	0	202	202	0	3	3	0: 0: 0	1: 1:14	1: 1:14
SOC	0	9	9	0	1	1	0: 0: 0	0: 3:23	0: 3:23
SRL	0	75	75	0	4	4	0: 0: 0	2:11:54	2:11:54
SWS	0	49	49	0	4	4	0: 0: 0	1:18:23	1:18:23
TAM	94	150	244	4	10	14	2:36: 5	1: 5:28	3:41:33
USGS	0	4	4	0	1	1	0: 0: 0	0: 3:53	0: 3:53
VPP	0	1	1	0	1	1	0: 0: 0	0: 0: 9	0: 0: 9
WPGU	0	5	5	0	1	1	0: 0: 0	0: 3:18	0: 3:18
ZOOL	22	25	47	1	2	3	0: 5:49	0:20:18	0:26: 7
SUBTOTAL	22255	7515	29770	111	294	405	121: 0:49	127:18: 3	248:18:52

DCSSYS[3]	0	3084	3084		24	24	0:00:00	91:37:23	91:37:23
OVRHED[4]	0	3471	3471		1	1	[illegible]:00:00	15:19:0	15:19:0
REFUND[5]	0	20	20	0	1	1	0:00:00	0:29:52	0:29:52
SSUAD[6]	0	56	56		1	1	0:00:00	1:18:56	1:18:56
XDCS[7]	0	106	106	0	3	3	0:00:00	52:56:31	52:56:31
XMAINT[8]	0	17	17	0	5	5	0:00:00	42:47:19	42:47:19
XSSS[9]	0	304	304	0	1	1	0:00:00	1:52:10	1:52:10
TOTAL	22255	14573	36828	111	33[illegible]	441	121:[illegible]:49	333:39:14	454:4[illegible]:[illegible]3

1 See list of departmental codes following
2 Training and Education
3 System Improvement and Modifications
4 System Use in Excess of User Estimated Usage
5 Rerun of User Jobs Due to Machine or System Error
6 University Administrative Overhead Use
7 System Updating
8 System Maintenance
9 Special Short Shots

Table I - IBM 360/20

Summary of Use

December, 1968

Unscheduled Engineering		54:26
Scheduled Engineering		4:40
Maintenance		3:26
Production		312:07
Operator Training		:15
Air-Conditioning		1:30
Idle		70:21
	Total	446:45

Table II - IBM 360/20

Summary of Machine Errors

December, 1968

2560 MFCM		27
2203 Printer		6
	Total	33

Table I - IBM 360/50-75

Summary of Use

December, 1968

Unscheduled Engineering	20:13:00		
Scheduled Engineering	34:58:06		
Maintenance	:13		
Idle	3:00:00		
		Subtotal	58:11:19
Total Use			
Training and Education		143:00:58	
System Improvement and Modifications		87:39:44	
System Updating		10:02:53	
University Administrative Overhead Use		32:18	
Overhead		21:06:01	
Refund		1:56:34	
Customer Use			
In System	114:50:37		
Special Short Shots	2:05:34		
Customer Use Total		116:56:11	
		Total Use	381:14:39
		Total Time	439:25:58

Table II - IBM 360/50-75

Summary of Errors

December, 1968

75 CPU		1
75 Core		1
50 CPU		2
50 Main Frame		2
2540 Card Reader Punch		1
1052 Console Typewriter		1
	Total	8

IBM 360/75 TABLE III DEC., 1968

DEPT[1]	NUMBER OF RUNS			NUMBER OF SPECS			IBM 360/75 USAGE IN HH:MM:SS		
	T AND E[2]	RES	TOTAL	T AND E[2]	RES	TOTAL	T AND E[2]	RES	TOTAL
AAE	600	190	790	5	5	10	5:50:54	6:43:21	12:34:15
ACCY	0	0	0	1	0	1	0: 1: 0	0: 0: 0	0: 1: 0
AGE	109	52	161	7	3	10	1:36: 3	0:44:12	2:20:15
AGEC	50	225	275	1	10	11	0:24:41	3:33:52	3:58:33
AGRON	303	150	453	1	5	6	2:34:24	1: 7:13	3:41:37
ANS	0	51	51	0	2	2	0: 0: 0	0:10:34	0:10:34
ARCH	162	7	169	3	1	4	1: 9:30	0: 2:55	1:12:25
ASTR	0	44	44	0	6	6	0: 0: 0	0: 5: 8	0: 5: 8
BECBSR	0	29	29	0	1	1	0: 0: 0	0:21:59	0:21:59
CE	2067	920	2987	15	39	54	12:53:47	18:18:52	31:12:39
CHE	384	771	1155	6	22	28	2:56: 5	13:45: 2	16:41: 7
CRC	0	17	17	0	1	1	0: 0: 0	0: 5:18	0: 5:18
CSL	0	9	9	0	1	1	0: 0: 0	0: 3:37	0: 3:37
DCS	6709	224	6933	17	12	29	75:44:36	8:58:12	84:42:48
DS	0	29	29	0	2	2	0: 0: 0	0:14:30	0:14:30
DUE	27	0	27	1	0	1	0: 3:51	0: 0: 0	0: 3:51
ECON	31	5	36	1	2	3	0:29:27	0:13:15	0:16:12
ED	0	38	38	0	3	3	0: 0: 0	0:24:56	0:24:56
EDPSY	29	0	29	2	0	2	0: 8:14	0: 0: 0	0: 8:14
EE	1915	551	2466	15	27	42	9:54: 1	4:36:43	14:30:44
ENGADM	0	42	42	0	2	2	0: 0: 0	0:41:49	0:41:49
ENGCST	0	36	36	0	1	1	0: 0: 0	0:27:39	0:27:39
ENGH	172	0	172	1	0	1	0:24:44	0: 0: 0	0:24:44
ENGRDV	0	0	0	0	1	1	0: 0: 0	0: 0:26	0: 0:26
FT	0	34	34	0	3	3	0: 0: 0	0:11:59	0:11:59

GENE	235	0	235	3	0	3	1:49: 6	0: 0: 0	1:49: 6
GEOG	0	107	107	0	2	2	0: 0: 0	0:41:51	0:41:51
GEOL	219	105	324	3	2	5	4: 5:56	0:27:23	4:33:19
GER	0	1	1	0	1	1	0: 0: 0	0: 2:23	0: 2:23
GSBA	586	0	585	3	0	3	3:10:18	0: 0: 0	3:10:18
ICR	0	62	62	0	2	2	0: 0: 0	0:49:51	0:49:51
IE	8	0	8	1	0	1	0: 2:53	0: 0: 0	0: 2:53
ILLDMH	0	118	118	0	1	1	0: 0: 0	6:14: 9	6:14: 9
INADM	0	18	18	0	1	1	0: 0: 0	0: 8:39	0: 8:39
MATH	0	4	4	0	2	2	0: 0: 0	0: 1:48	0: 1:48
MATRL	0	366	365	0	17	17	0: 0: 0	8:23:25	8:23:25
MCBIO	0	2	2	0	1	1	0: 0: 0	0: 0:50	0: 0:50
ME	2219	194	2413	18	8	26	14:28:41	3: 2: 5	17:30:46
MKTG	0	17	17	0	1	1	0: 0: 0	0: 6:41	0: 6:41
MMPE	0	0	0	0	2	2	0: 0: 0	0: 1:20	0: 1:20
MUSIC	0	7	7	0	1	1	0: 0: 0	0: 3:28	0: 3:28
NHS	0	0	0	0	1	1	0: 0: 0	0: 2:41	0: 2:41
NUCE	61	186	247	2	9	11	0: 9:44	2:15:39	2:25:23
OIR	0	52	52	0	1	1	0: 0: 0	0:20:55	0:20:55
ORME	0	12	12	0	1	1	0: 0: 0	0: 4:16	0: 4:16
PHYB	0	118	118	0	2	2	0: 0: 0	3:24:26	3:24:26
PHYCS	111	408	519	4	13	17	0:22:44	7:36:19	7:59: 3
PHYX	0	601	601	0	26	26	0: 0: 0	14:41:19	14:41:19
PLPA	19	0	19	1	0	1	0: 1:55	0: 0: 0	0: 1:55
POLS	26	24	50	1	1	2	0:21:22	0:11:37	0:32:59
PSYCH	0	48	48	0	3	3	0: 0: 0	0:33:41	0:33:41
REC	0	3	3	0	1	1	0: 0: 0	0: 1:16	0: 1:16
SCONS	0	65	65	0	1	1	0: 0: 0	0:23:46	0:23:46
SGS	0	201	201	0	1	1	0: 0: 0	0:48:59	0:48:59

SRL	0	72	72	0	5	5	0: 0: 0	0:48:14	0:48:14
SWS	0	142	142	0	6	6	0: 0: 0	2:39:43	2:39:43
TAM	154	41	195	4	8	12	4:13:17	0:28:30	4:41:47
USGS	0	10	10	0	1	1	0: 0: 0	0: 5:32	0: 5:32
VPP	0	4	4	0	1	1	0: 0: 0	0: 0:57	0: 0:57
WPGU	0	1	1	0	1	1	0: 0: 0	0: 0:55	0: 0:55
ZOOL	15	6	21	1	1	2	0: 5:45	0: 1:51	0: 7:36
SUBTOTAL	16211	6419	22630	117	273	390	143: 0:58	114:50:37	257:51:35
DCSSYS[3]	0	3044	3044	0	25	25	0: 0: 0	87:39:44	87:39:44
OVRHED[4]	0	3311	3311	0	1	1	0: 0: 0	21: 6: 1	21: 6: 1
REFUND[5]	0	45	45	0	1	1	0: 0: 0	1:56:34	1:56:34
SSUAD[6]	0	32	32	0	1	1	0: 0: 0	0:32:18	0:32:18
XDCS[7]	0	96	95	0	2	2	0: 0: 0	10: 2:53	10: 2:53
XMAINT[8]	0	31	31	0	4	4	0: 0: 0	58:11:19	58:11:19
XSSS[9]	0	360	360	0	1	1	0: 0: 0	2: 5:34	2: 5:34
TOTAL	16211	13338	29549	117	308	425	143: 0:58	296:25: 0	439:25:58

1 See list of departmental codes following
2 Training and Education
3 System Improvement and Modifications
4 System Use in Excess of User Estimated Usage
5 Rerun of User Jobs Due to Machine or System Error
6 University Administrative Overhead Use
7 System Updating
8 System Maintenance
9 Special Short Shots

IBM 360/75 TABLE III 4TH QTR, 1968

DEPT[1]	NUMBER OF RUNS			NUMBER OF SPECS			IBM 360/75 USAGE IN HH:MM:SS		
	T AND E[2]	RES	TOTAL	T AND E[2]	RES	TOTAL	T AND E[2]	RES	TOTAL
AAE	1845	423	2268	15	11	26	19:37: 5	13:26: 7	33: 3:12
ACCY	526	23	549	3	4	7	2: 2:11	0:24: 0	2:26:11
AGE	581	177	758	16	8	24	11:17:27	2: 2:41	13:20: 8
AGEC	50	712	762	1	28	29	0:24:41	13:28:54	13:53:35
AGRON	1252	544	1796	3	19	22	6:30:55	3:25:44	9:56:39
ANS	0	173	173	0	6	6	0: 0: 0	1:19:50	1:19:50
ARCH	515	8	523	8	2	10	3:10:53	0: 3:47	3:14:40
ASIR	0	230	230	0	16	16	0: 0: 0	1:51:36	1:51:36
BECBSR	0	71	71	0	3	3	0: 0: 0	0:55:36	0:55:36
CE	4475	2563	7038	40	117	157	30:50: 2	56: 9:22	86:59:24
CHE	1009	2298	3307	16	73	89	6:39:53	45:10:19	51:50:12
CRC	0	88	88	0	3	3	0: 0: 0	0:47: 6	0:47: 6
CSL	0	25	25	0	3	3	0: 0: 0	0:10: 0	0:10: 0
DCS	35390	725	36115	48	31	79	206: 9:50	27: 4:58	233:14:48
DS	0	57	57	0	4	4	0: 0: 0	0:35:18	0:35:18
DUE	61	0	61	3	0	3	0: 9:29	0: 0: 0	0: 9:29
ECON	31	10	41	1	4	5	0:29:27	0:18:32	0:47:59
ED	0	135	135	0	8	8	0: 0: 0	1:38:46	1:38:46
EDADM	0	35	35	0	1	1	0: 0: 0	0:54:40	0:54:40
EDPSY	149	0	149	6	0	6	0:47:56	0: 0: 0	0:47:56
EE	6377	1905	8282	43	88	131	40:37:34	19: 7:13	59:44:47
ENGADM	0	188	188	0	6	6	0: 0: 0	2:28:43	2:28:43
ENGCST	0	110	110	0	3	3	0: 0: 0	1:38:38	1:38:38
ENGH	774	0	774	3	0	3	2: 7:35	0: 0: 0	2: 7:35
ENGRDV	0	12	12	0	3	3	0: 0: 0	0: 3:50	0: 3:50
FI	0	69	69	0	8	8	0: 0: 0	0:26:42	0:26:42

GENE	855		855	11		11	5:22:31	0: 0: 0	5:22:31
GEOG		173	173		6	6	0: 0: 0	1:23:49	1:23:49
GEOL	421	149	570	9	5	14	5:35:14	0:50:13	6:25:27
GER	0	5	5	0	3	3	0: 0: 0	0: 3: 3	0: 3: 3
GSBA	1253	0	1253	9	0	9	7:50:26	0: 0: 0	7:50:26
HORT		15	15	0	2	2	0: 0: 0	0:20:20	0:20:20
ICR	0	226	226	0	5	5	0: 0: 0	2:44:29	2:44:29
IE	63	0	63	3		3	0:24: 7	0: 0: 0	0:24: 7
ILLCF	0	5	5	0	2	2	0: 0: 0	0: 2:26	0: 2:26
ILLDMH	0	629	629	0	3	3	0: 0: 0	21: 1: 5	21: 1: 5
INADM	0	38	38	0	3	3	0: 0: 0	0:15:59	0:15:59
MATH	0	39	39	0	5	5	0: 0: 0	0:18:23	0:18:23
MATRL		1641	1641	0	62	62	0: 0: 0	30:24:58	30:24:58
MCBIO		28	28	0	3	3	0: 0: 0	0:12: 4	0:12: 4
ME	6575	627	7202	52	26	78	34:39: 5	7:32:45	42:11:50
MKTG	0	27	27	0	2	2	0: 0: 0	0: 9: 3	0: 9: 3
MMPE	0	129	129	0	6	6	0: 0: 0	4:40: 7	4:40: 7
MUSIC		7	7	0	1	1	0: 0: 0	0: 3:28	0: 3:28
NHS	0	71	71	0	4	4	0: 0: 0	1:56:24	1:56:24
NUCE	193	523	716	8	24	32	0:53:43	5:34:50	6:28:33
OIR	0	353	353	0	3	3	0: 0: 0	3:11:51	3:11:51
ORME	0	24	24	0	3	3	0: 0: 0	0: 8:24	0: 8:24
PHYB	0	265	265	0	5	5	0: 0: 0	7: 3:58	7: 3:58
PHYCS	226	1620	1846	8	41	49	0:48:22	25:11:43	26: 0: 5
PHYX	0	2572	2572	0	59	59	0: 0: 0	54:48:41	54:48:41
PLPA	19	39	58	1	2	3	0: 1:55	0: 6: 5	0: 8: 0
POLS	58	57	115	3	3	6	1: 2:58	0:20:40	1:23:38
PSYCH	2	105	107	1	9	10	0: 0:37	1: 3:58	1: 4:35

REC	0	28	28	0	4	4	0: 0: 0	1:55:54	1:55:54
SCONS	0	162	162	0	3	3	0: 0: 0	1:49:35	1:49:35
SGS	0	643	643	0	6	6	0: 0: 0	2:59:16	2:59:16
SOC	0	12	12	0	2	2	0: 0: 0	0: 4:42	0: 4:42
SRL	0	258	258	0	12	12	0: 0: 0	0: 2:34	0: 2:34
SWS	0	256	256	0	15	15	0: 0: 0	5:50: 0	5:50: 0
TAM	253	317	570	9	26	35	6:51: 9	2:49:32	9:40:41
USGS	0	14	14	0	2	2	0: 0: 0	0: 9:25	0: 9:25
VPP	0	6	6	0	3	3	0: 0: 0	0: 1:13	0: 1:13
WPGU	0	8	8	0	3	3	0: 0: 0	0: 5:29	0: 5:29
ZOOL	100	121	221	3	8	11	0:23:22	1:14:47	1:38: 9
SUBTOTAL[3]	63053	21773	84826	323	820	1143	394:48:27	379:52:21	774:40:48
DCSSYS[3]	0	9687	9687	0	70	70	0: 0: 0	278:22:55	278:22:55
OVRHED[4]	0	1357[illegible]	1357[illegible]	0	3	3	0: 0: 0	54:14:16	54:14:16
REFUND[5]	0	76	76	0	3	3	0: 0: 0	3: 5:46	3: 5:46
SSUAC[6]	0	14[illegible]	14[illegible]	0	3	3	0: 0: 0	3: 2:24	3: 2:24
XDCS[7]	0	493	493	0	8	8	0: 0: 0	117: 4:22	117: 4:22
XMAINT[8]	0	62	62	0	13	13	0: 0: 0	160: 4: 7	160: 4: 7
XSSS[9]	0	942	942	0	3	3	0: 0: 0	5:41:50	5:41:50
TOTAL	63053	46743	109796	323	923	1246	394:48:27	1001:28: 1	1396:16:28

1 See list of departmental codes following
2 Training and Education
3 System Improvement and Modifications
4 System Use in Excess of User Estimated Usage
5 Rerun of User Jobs Due to Machine or System Error
6 University Administrative Overhead Use
7 System Updating
8 System Maintenance
9 Special Short Shots

TABLE III

November-December, 1968

IBM 1800

DEPT	NO. OF RUNS			USAGE IN HOURS.MINS.SECS.		
	T and E	RES.	TOTAL	T and E	RES.	TOTAL
EE	0	4	4	0	17.12	17.12
DCS	Dec. 81	0	81	01.33.28	0	01.33.28
SUBTOTAL	81	4	85	01.33.28	17.12	01.50.40
DCSSYS	0	97	97	0	04.37.46	04.37.46
SUBTOTAL	0	97	97	00.00.00	04.37.46	04.37.46
TOTALS	81	101	182	01.33.28	04.54.58	06.28.26

9.2 Research Problem Specifications

During the fourth quarter of 1968, 143 problem specifications were submitted to the Department for computation on the System/360. The following brief descriptions of these problems have been prepared for inclusion in this report by those submitting them. T indicates a calculation associated with a thesis.

1908 Theoretical and Applied Mechanics. Finite Element Analysis of Concrete Stresses. The finite element analysis is used to predict shrinkage stresses in reinforced concrete. (H. Rejali)

1909 T Mechanical Engineering. Pulsating Flow of Water. Pulsating flow across orifices. (Thomas Sytko)

1916 Agricultural Economics. Optimal Irrigation Plans. Programs will be developed and solved for selected Illinois farm situations involving irrigation investments. Coefficients, objective function weights, and right-hand sides will be varied in sensitivity analysis to determine those factors most important in determining an optimal investment in irrigation. (E. R. Swanson)

1920 T Computer Science. Separable Boolean Functions. The set of Boolean functions can be classified by three different separability criteria. Thus enumeration of properties of a separable, pseudo-separable, and marginally separable switching functions will be conducted. (Baugh)

1921 T Civil Engineering. Iterative Design Analyzer. Analyzer suitable for iterative design of skeletal structures. (R. Ertas)

1922 Physics. Neutron Resonance Analysis. Continuation of previous research on neutron resonance analysis, aiming toward a program suitable for simultaneous correlated analysis of fission, capture, and total cross sections. (F. T. Adler)

1923 Coordinated Science Laboratory. Cognitive Memory Project. Various data bases will be processed to be used as input to the computer facilities at CSL. The basic problem is to design a sophisticated question-answering system based on natural language input/output. (R. T. Chien)

1924 T Materials Research Laboratory. Data Analysis for Thermoelectric Power. This computer time will be used for least squares analysis of thermoelectric power data. (C. M. Wayman)

1926 Mathematics. Design Construction. Combinatorial designs are constructed by enumerating certain sub-designs and using a computer to attempt a completion to a full design. (E. T. Parker)

1929 Civil Engineering. Study of the Hydrology for Models of the Great Lakes. Quantitative analysis of precipitation, evaporation, and runoff for each of the Great Lakes and the Great Lakes as a system are to be made in order to provide a better understanding of these relationships for future management of the water resources of the Great Lakes. (D. Meredith)

1931 Natural History Survey. Mineral Content of Bird Feathers. The identification of goose nesting grounds by trace element analysis of wing feathers. (Harold C. Hanson)

1932 Agricultural Economics. Price Analysis. An analysis will be done on the price structures in the agricultural field. (Raymond Leuthold)

1933 Economics. Negative Income Tax. A study of various income maintenance plans such as the negative income tax, their effects on incentives, their impact on the economy, their estimated cost. (Husby)

1934 T Mechanical Engineering. Effects of Torsional and Axial Vibrations on Frictional Losses in Gear Systems. To investigate the effects of axial and torsional vibration on frictional losses in gear systems. (Mamoun)

1935 Economics. Quadratic Programming Estimators and Specification. This time will be used to continue work on the Monte Carlo experiment for quadratic programming estimators which we have been doing under problem specification number 63044 on the 7094. (T. A. Yancey)

1936 Astronomy. Distances of Planetary Nebulae and Galactic Structure. The use of measured extinction will serve to establish position of spiral arms of galaxy. This in turn leads to more precise determination of local space densities of planetary nebulae. (Julius H. Cahn)

1938 Electrical Engineering. Research on a Broadband Isotropic Antenna System. This problem involves research aimed at the development of a configuration of antennas which will radiate a signal equally well in all directions. Machine time will be used in the analysis and design of the necessary antennas. (Dyson)

1939 Mechanical Engineering. IBR Steam and Water Heating Research. This is a continuing research program on hydronic heating and cooling. The computer is used for data analysis and plotting. (W. S. Harris)

1940 Animal Science. Statistical Genetics of Lamb Meat Production. Data processing and statistical analysis for an extensive sheep breeding project to evaluate heritabilities and genetic correlations for many traits related to lamb meat productivity. (H. W. Norton)

1941 Civil Engineering. Numerical Analysis of Cable Structures. This program deals with an exact non-linear solution of suspended cable networks, subjected to statically applied loads. The netword is treated as comprising discrete cable elements. (Prem Krishna)

1942 T Geography. Climatic Fluctuation: A Case Study in Precipitation Analysis. This study will be a detailed analysis of precipitation time series using spectrum and cross-spectrum analyses for various stations in Illinois. The data will also be used in part for the generation of a simulation model of precipitation events and in part for the testing of the model. (John Lewis, Jr.)

1943 Geology. Microprobe and X-ray Diffraction Mineral Analysis. The geochemical evolution of the minerals is being studied in a group of related igneous rocks from New England. The compositional changes in the evolution of the mineral phases are being followed by microprobe analysis of individual mineral crystals. The microprobe output has to be corrected by computer in order to obtain accurate compositions. The structural states and relations of unit cells to composition are being determined by high-precision X-ray diffractometry. Accurate cell parameters are being obtained by a self indexing least squares refinement program. (D. M. Henderson)

1944 T Mechanical Engineering. Tandem Airfoils in a Diffuser. This problem is the study of the flow of air past two airfoils in tandem with the pressure generally increasing in the flow direction. The effects of boundary layers and wakes are included in this two dimensional problem. Numerical methods are to be used in the analysis. (Morgan)

1946 Electrical Engineering. Oxygen-Hydrogen Atmosphere. The diurnal variation of important oxygen and hydrogen constituents in the atmosphere between 60 and 90 km is being studied. (Stowe)

1947 Mechanical Engineering. Thermoregulatory Systems. Mathematical evaulation of various aspects of thermal control of a protective suit for humans. (Chato)

1948 Theoretical and Applied Mechanics. Deflection of Plates. Energy solution, collocation solution, and complex variable solution for deflection of clamped plates of a general type. (W. J. Worley)

1949 Economics. Optimal Financial Structure and Investment Expenditures. This project will theoretically determine the optimal financial structure for a business firm and empirically find the effect of these results on investment expenditures by firms. In the theoretical part of the work numerical integration is required and will be performed on the 360. Following this the empirical estimation will be performed using techniques which allow a wide variety of lag structures while including appropriate financial variables. (Resek)

1951 Aeronautical and Astronautical Engineering. Structural Dynamics. Contract research in the area of undamped and damped structural dynamic problems. (Zak)

1952 T Electrical Engineering. Mimimax Approximation with Rational Functions. The project is concerned with determining poles and zeros of transfer functions such that a specified amplitude response may be realized, in the sense of a minimax approximation. This requires approximation with rational functions, resulting in systems of non-linear simultaneous equations. A computer program for the solution of this approximation problem will be developed. (W. P. Petzold)

1954 Mechanical Engineering. Suspension Boundary Layer Flow. The time requested will be used for short computational problems for turbulent flow of suspensions over parallel plates. (J. J. Stukel)

1955 Animal Science. Lipid Metabolism in the Beef Animal. Detailed study of lipid metabolism in the beef animal to determine: (1) the means by which a beef animal deposits lipid within certain depots in its body during growth and development; (2) the mechanisms of lipid mobilization, transport and utilization; and (3) the influence of physiological status as mediated by hormonal control of items 1 and 2. (John R. Romans)

1956 Electrical Engineering. Full Wave Solutions. To theoretically determine the reflection coefficients for obliquely incident LF and VLF waves reflecting from the D region of the ionosphere, given electron density and collision frequency profiles, magnitude and direction of the earth's local magnetic field, angle of incidence of the waves, frequency of the waves, upper and lower limits for the downward numerical integration, integration interval, and desired accuracy. (Viertel)

1957 T Chemistry and Chemical Engineering. Flow Through Regular Packed Beds. An attempt is made to get some understanding of the flow patterns in a regular packed bed by measuring local properties such as mass transfer coefficients and shear stress. Also by means of an "analog computer" values of the velocity for inviscid flow are obtained. The digital computer must be used to fit a polynomial to the experimental data. (Anastas Karabelas)

1959 Geology. X-ray Analysis. X-ray analysis of mineral structures involves a large amount of data collecting and subsequent processing. Mathematical methods like least-squares analysis and FOURIER synthesis are extensively used. A series of programs in 360/FORTRAN will be kept on disk files together with the collected data. (Necip Guven)

1960 Plant Pathology. Field Corn Disease Research. Project involves evaluation of lines and varieties of field corn as sources of disease resistance and genetic studies of host resistance. Appropriate statistical and genetic, quantitative and qualitative, procedures will be used to analyze results from inheritance and evaluation studies, and to predict performance of disease resistant lines and hybrids. (Hooker)

1961 Computer Science. Tracing Program. The purpose is to adapt TRACK, a FORTRAN program for tracing the behavior of other FORTRAN programs, to the IBM 360 operating system and to investigate the feasibility of a similar code for ALGOL. (Fosdick)

1962 Coordinated Science Laboratory. Cognitive Memory System Simulation. A program will be coded to simulate cognitive memory. (Benjamin Wang)

1964 Materials Research Laboratory. Effect of Acoustical Phonons in Si on Tunneling. The contribution of acoustical phonons to the electron-self energy is calculated, with this function

$$\Sigma(p,E)$$

then the second derivative of the tunneling current (d^2I/dv^2) is calculated for Pb-Si tunnel junctions. (Heinz J. Deuling)

1965 Electrical Engineering. Rocket Data Processing. The Aeronomy Laboratory's Rocket Program includes the analysis of Faraday rotation and differential absorption data from Nike Apache rocket experiments in the D-region of the ionosphere. The processing of these data involves the Sen-Wyller equations of the generalized magnetoionic theory and a representation of the geomagnetic field by spherical harmonic expansions. This work can best (and perhaps only) be done by digital computer. (E. A. Mechtly)

1966 Food Science. Plot Mass Spectral. Normalize and plot mass spectral data from cards. (E. G. Perins)

1967 Industrial Administration. Team Decision Problems. Stochastic linear programming computations of management control problems. For creating input data to the above computations we need to write several programs. Data thus created are computed through the linear programming program available in MPS. (H. Hinomoto)

1968 Architecture. Economics in Building Structures. Faculty research on comparative framing of building frames. (Brightbill)

1969 T Materials Research Laboratory. Curve Fitting. Problem will be to fit curves associated with Mossbauer spectroscopy. (H. G. Drickamer)

1972 T Electrical Engineering. High Voltage Direct Current Transmission. An economic evaluation of a high voltage direct current transmission line. (Stephen R. Lambert)

1973 Agronomy. Consumer and Marketing Service Research. Program development for Consumer and Marketing Service Inspection and marketing problems. (Swanson)

1974 T Theoretical and Applied Mechanics. Domain of Possible Deformation of an Hyperelastic Material of Grade One. There are various interrelationships among the invariants of deformation right Cauchy-Green tensor. This project will use these interrelationships to determine the domain of possible deformation in the invariant space. (T. Moriarty)

1975 T Architecture. Low Cost Suburban Chicago Housing. The development of a hierarchial semi-lattice subset structure for a group of elements whose organization is defined by a non-directed graph. (Robert Katz)

1977 Mechanical Engineering. Finite-Difference Solution of Parabolic Equations. A study of finite-difference solutions of the diffusion equation and the boundary layer equations. (A. M. Clausing)

1978 T Recreation. Analysis of Water Resources Recreation Literature. The IBM 360 Document Processing System will be used to enable this investigator to receive, using key word inputs, those bibliographical notations that are pertinent to his request. This bibliographic retrieval process will be used to classify according to a topical outline model pieces of literature providing insight into concepts and scientific results of research developed in numerous allied disciplines. A "State-of-the-Art" paper will be developed which will deal with and illuminate the most apparent relationships involved in the recreational aspects of water resources use, planning and development. (R. B. Ditton)

1979 T Mechanical Engineering. Non-Constant Pressure Mixing. A study of the velocity profiles in the mixing region of two gas streams under the influence of a streamwise pressure gradient. (D. F. Brink)

1981 T Theoretical and Applied Mechanics. Scattered Light Photo-elasticity. The problem is to develop a method for determining stress distributions by means of scattered light photoelasticity. This project is to improve on experimental procedures and to incorporate a more systematic means of data reduction. (Martin)

1984 T Theoretical and Applied Mechanics. Concrete Columns Reinforced with High Strength Steels Subjected to Biaxially Eccentric Loading. A theoretical and experimental analysis of the behavior, ultimate strength and mode of failure of biaxially eccentrically loaded rectangular concrete columns with corner reinforcement of high strength steels. (P. D. Heimdahl)

1985 Chemistry and Chemical Engineering. Stability of Falling Film. Compute eigenvalues for stability of laminar liquid film flowing down inclined wall. (Jameson)

1986 State Geological Survey. Mineral Matter in Coal. Size analyses of coal and of mineral matter obtained from coal are being done on a "Coulter Counter". Data reduction by computer will assist in determining ffects of size of mineral particles in coal on problems of air pollution. (H. Gluskoter)

1987 Geography. Soil Toxicity Trends in Southern California. In this investigation, an attempt will be made to evaluate the spatial relationships between soil toxicity patterns, topography (using slope and topographic position parameters) soil texture, soil cation exchange capacity, and geologic fault patterns in Castac Valley California. The spatial nature of soil toxicity patterns will be analyzed by spectrum analysis (in a two-dimensional spatial model rather than a time series model). The relationships between soil toxicity and the other environmental parameters will be studied in a series of cospectrum analyses. (Placido LaValle)

1988 T Civil Engineering. Time Dependent Closure of a Spherical Cavity. Analysis of stress-redistribution and creep closure of a spherical cavity in an infinite creep-sensitive material. (Aiyer)

1989 T Theoretical and Applied Mechanics. Numerical Analysis of Plane Stress Problems with Creep. Evaluation of flow criteria and constitutive relations for the numerical analysis of plane stress problems with creep using finite element techniques. (T. Pickel)

1990 T Nuclear Engineering. Solution of Diffusion-Depletion Equations for Fast Breeder Reactors. Solution of coupled diffusion-depletion equations which describe the breeding characteristics of large fast breeder reactors. (Philbin)

1991 T Civil Engineering. Plane Structure-Medium Interaction. A study to evaluate the structure-medium interaction due to a plane elastic stress wave generated from dynamic disturbances. (E. G. Tabujara)

1992 T Civil Engineering. Pavement Optimization. A study directed toward the establishment of principles of guidelines that can be used to promote the most effective use of local and stabilized materials in the construction of low traffic highway pavements. (Q. L. Robnett)

1993 Civil Engineering. Dispersion in Unsteady Flow. Prior analytical word on dispersion has dealt primarily with steady flow conditions. It is now possible to use numerical simulation for analytical study of dispersion in unsteady flow problems. Specifically, it is planned to use numerical simulation to investigate (1) the influence of flood hydrographs on dispersion in streams and (2) the influence of tidal characteristics on dispersion in estuaries. These are problems of significance in current engineering practice. The planned studies would help to delineate the significance of various physical parameters. (Holley)

1994 Civil Engineering. Oxygen Demand and Reaeration. Chemical and biochemical oxygen demand in turbulent water can influence the net rate of oxygen absorption from the atmosphere. The computer will be used to evaluate analytical solutions for absorption and diffusion rates and to perform the statistical analysis of laboratory data. (Holley)

1995 Civil Engineering. Reinforced Concrete Arches. The behavior of concrete arches under transverse load from an unloaded stage to collapse is studied using a numerical procedure to evaluate the effects of variation in the following parameters: 1) concrete strength; 2) yielding stress of steel reinforcement; 3) percentage of reinforcement; and 4) ratio of midspan rise to span of the arch. (German Gurfinkel)

1996 T Agricultural Economics. Plant Location Problem. Using a combinatorial approach with the bounded algorithm of linear programming, I would like to find out the minimum-total-cost solution about the processing plant location, number, and size when the plant operation has a chracteristic of the "economies of scale." (Hiroshi Tsujii)

1997 Survey Research Laboratory. Test Runs - Tabs. Test runs before analysis of survey results. (M. F. Uchida)

1998 Survey Research Laboratory. PL/1 - Development. Development and conversion of PL/1 for survey purposes. (M. F. Uchida)

2001 Educational Administration. Basic Information Retrieval System (BIRS). The Education Resources Information Clearinghous (ERIC) on Early Childhood Education is required to establish an internal document retrieval system. This project will develop a number of computer programs (BIRS) which will automate the clerical operations common to most information retrieval systems, e.g., file maintenance and searching. (Carss)

2002 T Nuclear Engineering. Time Studies of Plasma Interactions and Emissions. Plasma interactions and emissions are studied in time as particles are injected radially inwards in cylindrical and spherical geometries, solving simultaneously the equations of poisson, energy conservation, and charge continuity. (Dolan)

2004 T Civil Engineering. Solution of Consolidation Problems Using Finite Differences. Finite difference solution for the time-settlement relationship in a consolidation problem involving a system of contiguous, compressible soil layers with partially draining boundaries. (Fred Gau)

2005 T Chemistry and Chemical Engineering. Stability of Liquid Surface Under Electric Field. Numerical determination of voltage required to initiate instability on liquid surface. (Young Y. Kim)

2006 Computer Science. ILLIAC IV. Use of the IBM 360 to work on ILLIAC IV problems. (Hui-Wen Chen)

2007 Electrical Engineering. Waves in Plasma. This problem investigates dispersion surface for E.M. waves in plasma. Essentially, this involves solving a fourth order equation. The object is to give a computer graphic display of this dispersion surface. (Y. T. Lo)

2009 T Electrical Engineering. Electromagnetic Boundary Value Problems. Change time from 7094 problem specification 6N004 to 360/75. (Imbriale)

2010 Chemistry and Chemical Engineering. Microwave Spectroscopy. Calculation of rotational, quadrupole, and related effects in microwave spectroscopy. (Arthur C. Ferguson)

2012 Physics. Properties of Many Particle Systems. Consideration of properties (low temperature) of many body systems. (Gordon Baym)

2013 Theoretical and Applied Mechanics. Hemodynamics. This is to change 7094 problem specification 7Ø082 to the 360. (M. E. Clark)

2014 T Chemistry and Chemical Engineering. Experimental and Theoretical Study of Acid-Base Interactions. Fitting experimental heats of acid-base interactions to a two-parameter equation. (R. S. Drago)

2015 Civil Engineering. Recursion Adjustment of Coordinates. From a drawing the approximate coordinates of all the points can be scaled. (In three dimensions, if required) certain constraining conditions can now be stripulated for the points, e.g., collinearity, perpendicularity, etc., and emposed by a stepwise recursion adjustment to obviate the need for large matrix inversions. (L. A. White)

2016 T Accountancy. Biayesian Treatment of a Priori Information Using Monte-Carlo. Comparative study of different estimation techniques using Monte-Carlo technique. (Saadia Montasser)

2017 Geography. Morphological Variation of Karst Topography in Jamaica. Spatial variations in karst depression occurrence and form will be analyzed with respect to spatial variations in geolgic structure, hydrologic conditions, and lithology within a portion of Wertern Jamaica. The spatial variations of karst depressions will be examined in a two-dimensional spectral analysis of terrain surface configuration, and the influence of structure, hydrology, and lithology will be examined through a series of cospectrum studies. (La Valle)

2018 Business Administration. Location Models. Mixed integer programming problems in location. (Sovereign)

2019 Physics. Solid State Thermodynamics. Reduction of experimental data and curve fitting related to thermodynamic measurements of superconductors and ferromagnetic transitions. (D. E. Mapother)

2021 Civil Engineering. Gradually Varied Flow. This replaces 7094 problem specification 76032. (Cheng-Lung Chen)

2023 Chemistry and Chemical Engineering. X-ray Structural Studies of Phosphonitriles. Crystal structure studies are being carried out on a number of derivatives of phosphonitrilic halides. The x-ray method is a powerful and unique tool to establish the structures of inorganic molecules. The methods being used are mainly direct methods programs. The computations involve structure factor calculations, formation of Fourier series, and refinement by least squares techniques. (Arrington)

2024 T Electrical Engineering. Electron Multiplier Design. Design of an electron multiplier device employing static electric and magnetic fields with high frequency electric fields to permit operation with microwave bandwidth. (Schaefer)

2026 T Civil Engineering. Drag Forces. Aim is the analytical determination of drag forces on bodies moving through non-Newtonian fluids. This is to be done via calculation of the energy dissipation by numerical methods. (Hormoz Pazwash)

2027 State Geological Survey. Steady State Ground-Water Flow Model. Finite difference techniques are applied to the solution of a 3D, non-homogeneous, isotropic flow system with the aid of a digital computer. (Paul Heigold)

2028 T Civil Engineering. Irregular Slab Analysis. A program entitled "A General Computer Program for the Elastic Analysis of Irregular Slabs" has been written by D. D. Moss which runs on the 7094. This request for computer time is made in order to modify this program so that it can be used on the 360. Once this has been accomplished the program shall be used to obtain data for a thesis concerning the study of slab behavior. (Charles H. Hofmayer)

2031 Theoretical and Applied Mechanics. Deflection of Curved Beam. The large and small deflections of a curved beam of variable cross section are calculated by successive summation of the deformations of a finite number of sections, each of which is considered to be circular. The beam is loaded by a force and/or a moment at each end. (J. C. McWhorter)

2032 Astronomy. Processing Photographic Data of Gaseous Nebulae. Photographic plates of gaseous nebulae are processed in the microphotomer. The digitized data (plate-transmission and plate-position) are recorded on magnetic tape which serves as input to the computer. The final output will include contour maps (= CalComp plots) of the distributions of the surface brightnesses across each nebula. Firstly, with the original resolution of the plate and, secondly, with a lower resolution. The output will also include the matrices of the surface brightnesses. (Helene Dickel)

2034 Children's Research Center. Production of KWIC Information Retrieval Project. The provision of a research tool by use of the computer to generate a master and KWIC index to 7,000 bibliographic entries considered relevant to research in the area of the motor performance of children. (M. J. Ellis)

2035 Children's Research Center. Production of KWIC Information Retrieval Project. The provision of a research tool by use of the computer to generate a master and KWIC index to 7,000 bibliographic entries considered relevant to research in the area of the motor performance of children. (M. J. Ellis)

2036 T Political Science. Post-Independence Electoral Politics in Ceylon. The major calculations will be correlations of data from six Parliamentary elections with delimitation and census statistics, and the correlation of party membership with other relevant characteristics of Members of Parliament. (W. W. Morris)

2038 T Political Science. Coalition Building. A simulation of presidential coalition building process using decision rules which relate to coalition size, ideological distances between delegates and candidates. (J. Zais)

2039 Veterinary Pathology and Hygiene. Endogenous Development of Laboratory Coccidia. Study of the life cycle including endogenous stages of Eimeria stiedae of rabbits. (Fitzgerald)

2040 T Psychology. Multidimensional Scaling. Multidimensional scaling with the direct solution of coefficients of a general distance function. (Pieszko)

2041 T Home Economics. Effect of Formulation on the Quality of Rice Flour Cake. This study is concerned with the quality of rice flour cake as affected by fat source, flour level, and carboxymethylcellulose gum addition. (Lsiekiss)

2043 Chemistry and Chemical Engineering. Line-Shape Calculations for NMR Adiabatic Fast Passage Experiments. These differential equations will be solved for a variety of experimental conditions for the Adiabatic fast passage. (J. Jonas)

2044 Physics. Analysis of Fissile Element Cross Sections. The work proposed here has as its basic aim the development of a group of computer programs which should become useful for the large scale analysis and evaluation of neutron resonance cross sections at Brookhaven National Laboratory and possibly at other installations or by experimental groups from which the data originate. Specifically, work in the following area is proposed: (1) An extension of the existing programs TANDEM and CODILLI, to provide for the simultaneous fitting of fission, capture, and total cross sections, or of any pair of cross sections; special attention will be devoted to make the codes usable for a range of resolution functions, and to treat, for example, transmission data with more than one sample thickness in an efficient manner. (2) Development of a special program, based directly on the use of complex arithmetic, to determine directly the complex partial fission widths, as discussed in Section III. (3) Study of recent developments concerning the procedures for accelerating convergence in least square or other optimization techniques, and implementation of useful devices in the programs. (4) Fitting of selected experimental data, which are of current interest and where the evaluation could complement other current work at BNL, ORNL, or other installations. (F. T. Adler)

2045 Astronomy. Eclipsing Binary Spectroscopy. The computer will be used to make linearization of microphotometer tracings of astronomical spectra of eclipsing binary stars. The CalComp plotter will be used to make intensity tracings. Line broadening calculations will also be made. (Edward Olson)

2047 T Theoretical and Applied Mechanics. Analytical Analysis of Statically Indeterminate Structures Made of Hollow Box Sections. Analysis of a series of statically indeterminate structures composed of hollow box sections to determine the reactions on the structures at fixed sections and stresses throughout the structures. (F. M. Thomas)

2048 T Agricultural Economics. Cattle Supply Study - Argentina. Multi-period linear programming about the cattle supply in Argentina. (Jose Kohout)

2050 Civil Engineering. Flow in Alluvial Rivers. Numerical analysis and data processing of resistance to flow in alluvial rivers. (Liou Ying-Chang)

2051 Civil Engineering. Watershed Runoff. Numerical analysis and data processing for investigations on the non-linearity and non-uniqueness of the unit hydrograph in the study of watershed runoff. (Anderson)

2052 Astronomy. Ionization Model of Orion Nebula. The Orion Nebula is a large mass of gas illuminated by a hot star. A mathematical model has been developed to predict the ionization structure of the nebula. The computer will be used to quantitatively evaluate the model parameters. (Julie Lutz)

2052 Bureau of Institutional Research. Administrative Research. The job number will be used for analysis of staff at the university in such areas as salary and length of service. Usage will also be made for analysis of student data such as curriculum, age, etc. (Walter Tousey)

2054 Astronomy. Studies in Gaseous Nebulae. The computer will be used for the reduction of nebular spectra and for the evaluation of nebular parameters from these spectra. (James Kaler)

2055 Materials Research Laboratory. Electron Scattering in Metals. The electrical resistivity and the quadrupole effects in NMR, both arising from the scattering of conduction electrons in metals, will be calculated by the pseudo-potential theory. (Y. Fukai)

2057 T Agricultural Economics. Monte Carlo Experiments. I am going to do some Monte Carlo experiments to study the small-sample properties of various systems of simultaneous equations with particular emphasis on forecasting and decision-making in an economic context. (Thomas Bell)

2058 T Aeronautical and Astronautical Engineering. Transverse Waves in Detonations. An attempt to determine the effect of a boundary layer on transverse waves in detonations. Using an acoustic approximation, a ray tracing technique will be used. (Webb)

2059 Civil Engineering. Constraint Processor. This program will be used to check a structure against the building codes. (Khan)

2061 Materials Research Laboratory. Research in Transition Metals and Alloys. Low temperature investigations of specific heats and magnetic susceptibilities of transition metal alloys. (P. A. Beck)

2062 Civil Engineering. Simulation of Corps of Engineers. Develop integrated series of operators to be used in a selected order to simulate particular decision-making processes on operations inherent in the work performed in Corps of Engineers. (L. R. Shaffer)

2063 T Civil Engineering. Competitive Strategy Models. Develop integrated series of competitive strategy models for comparative study of models utilizing data obtained in professional practice. (T. W. Micheau)

2064 T Civil Engineering. Calendar-Day CPM. Extend Calendar-Day CPM algorithm to include treatment of resources. (L. R. Shaffer)

2065 T Electrical Engineering. Sound Propagation Between Partially Connected Spaces. Theoretical calculations of sound transmission from one office area to another when the areas are separated by a "partial" partition (one that does not go all the way to the ceiling). (Schomer)

2066 Forestry. Soil Moisture Stress. To investigate the effects of controlled soil moisture stress on (1) foliar nutrients, number of growth flushes, diameter growth, and height growth of loblolly pine trees; (2) the specific gravity, tracheid length, and percentage of summerwood, springwood, cellulose, extractives, and lignin of the wood of loblolly pine trees. (A. R. Gilmore)

2068 Computer Science. Consulting. Automatic processing of test data for consulting work. (Gear)

2069 T Mining, Metallurgy, and Petroleum Engineering. Statistical Model of River Meanders. A spectral analysis of actual river meanders and statistically generated meanders is to be performed. (A. E. Scheidegger)

2072 Astronomy. Electronic Velocity Distributions in Stellar Atmospheres. Determination of stellar emission as dependent upon the functional form of the electronic velocity distribution function. (Julius H. Cahn)

2075 Psychology. Determination of Number of Factors. Determination of an objective decision rule for the number of factors to be extracted from a given correlation matrix. (L. G. Humphreys)

2077 T Theoretical and Applied Mechanics. Creep. Creep of load carrying members made of metals subjected to nonproportionate load changes. (S. Chu)

2078 T Electrical Engineering. Faraday Rotation D.A. Digitize two taped analog rocket signals using the 1800 ADC facility. Derive a FORTRAN program to extract Faraday Rotation as a phase shift using the sampling time base of the 1800, with the ADC output as suitable input. (A. G. Slekys)

2079 Physics. Optical Constants and Dispersion Relations. This problem is the same as problem specification 1436. (F. C. Brown)

2084 T Physics. Evaporation of Nuclear Particles. The thesis project includes numerical integration of a nuclear evaporation formula and analysis of experimental data testing nuclear evaporation theory. (William Switzer)

2085 T Theoretical and Applied Mechanics. Wave Propagation. Study of the effect of nonlinear boundary conditions on the propagation of waves in linear systems of finite length. Involves the numerical solution of hyperbolic partial differential equations with nonlinear boundaries. (Richard D. Strunk)

2086 Electrical Engineering. New Aspects of Antenna Synthesis. The objective of this problem is to search for a proper reactive reflecting surface, such as corrugated structure, so that the imaging property of the antenna system can be greatly improved. (Y. T. Lo)

2087 T Coordinated Science Laboratory. Tunneling Current Calculation. Solution of tunneling equation for metal-insulator-metal junctions. (Steven Depp)

2088 Nuclear Engineering. Coupled Core Studies. Preliminary studies of the transient behavior of two coupled critical reactors taking into account negative temperature feed back, using the code WIGLE. (G. Thayer)

2089 Agricultural Engineering. Data Analysis for Farm Waste Management Procedures. Data from farm animal waste treatment systems will be analyzed and compared using the 360/75 computer and the CalComp Plotter. (Don D. Jones)

2091 T Chemistry and Chemical Engineering. One-Center Hartree-Fock Calculations. The molecular Hartree-Fock equations are expanded in a series of numerical radial functions and spherical harmonics at one nucleus and the resulting 2nd order differential equations are solved numerically for the radial functions and eigenvalues by an iterative method. (R. L. Belford)

2092 State Water Survey. Radar Hail Evaluation. This project will involve analyses of various historical hail data to determine the most meaningful statistical seeding design. Studies of various past and recent surface hail data collected in Illinois to provide basic statistics on hailstorm characteristics. Developing a recording hailgage that would provide objective measures of hailfall intensity for employment in areas of modification projects. Radar-hail investigations to determine the capability of radars to detect storms producing surface hail in a modification project. (Robert Sinclair)

2094 Civil Engineering. Pavement Testing. Statistical analysis. (E. Barenberg)

2095 Provost's Office. Organizational Study of Departmental Performance. This project is an investigation of organizational factors of university academic departments and how these factors relate to various departmental outputs, such as quality of graduate students and publications of the faculty. (Lanier)

2097 T Physiology and Biophysics. Metabolic Hormonal Research. Routine isotope calculations, statistical analyses, curve fitting and plotting, and enzyme kinetical analyses for biochemical data. (Peter M. Spooner)

2099 Finance. Risk Simulation Model. To simulate a frequency distribution of benefit-cost ratios for a given investment decision. (Le Bloch)

2100 Children's Research Center. Analysis of Children's Play Behavior. The analysis of various physical and social factors as they interact in conditioning the play of preschool children. Locomotor behavior described by the changes in position in the Motor Performance and Play Research Laboratory will lead to emphasis concerning the effect of the physical environment on the activity patterns of the children at play. Analyses of the social structure of the play group will be undertaken to determine the interaction between social physical factors on play. (M. J. Ellis)

2101 T Chemistry and Chemical Engineering. Semi-Empirical Molecular Orbital Calculation. Application of semi-empirical molecular orbital theory to the description of carbonyl complexes. (T. L. Brown)

2102 T Civil Engineering. A Post-Buckling Analysis of Some Selected Structures. The set of nonlinear ordinary differential equations which describe the equilibrium of the structure at some load level are solved by a numerical procedure. The nonlinear equations are linearized analytically and the linear equations solved as a series of initial value problems by a predictor-corrector numerical integration process. Convergence of a sequence of these linearized problems to the solution of the nonlinear equations is sought through the application of Newton's method. (J. F. Harris)

2103 Education. Estudio de Problemas Intercultural. To develop and use natural language analysis programs for use in breaking down and integrating field notes gathered in the course of an examination of problems associated with differences in the cultures of the home and school. (Burnett)

2104 Agricultural Economics. Prepare Matrices for Linear Programming Problems. Formulation of linear programming problems for solution on the 360. (Baker)

2105 Electrical Engineering. Modeling the Plasma in a Discharge Tube. An attempt to model the plasma in a discharge tube on the basis of the external scattered fields. (Mittra-Schaubert)

2107 Bureau of Institutional Research. Administrative Research. Study of characteristics and activities of the University's staff and related variables. (Franklin L. Duff)

9.3 <u>Class Problem Specifications</u>

During the fourth quarter of 1968, 58 problem specifications were submitted to cover all assigned problems on the System/360 in the following courses.

1906	Electrical Engineering 416.
1907	Electrical Engineering 386.
1910	Computer Science 101.
1911	General Engineering 393.
1913	Electrical Engineering 386.
1914	Electrical Engineering 416.
1915	Civil Engineering 220.
1917	Physics 107.
1918	Computer Science 109.
1919	Electrical Engineering 382.
1925	Mechanical Engineering 495.
1927	Electrical Engineering 232.
1928	Electrical Engineering 383.
1930	Civil Engineering 352.
1937	Economics 476.

1945 Agricultural Engineering 299.

1950 Aeronautical and Astronautical Engineering 351.

1953 Computer Science 490.

1958 Aeronautical and Astronautical Engineering 296.

1963 Mathematics 123.

1970 Agricultural Engineering 499.

1971 Agricultural Engineering 287.

1976 Educational Psychology 497.

1980 Architecture 493.

1982 Physics 386.

1983 Mechanical Engineering 293.

1999 Civil Engineering 315.

2000 Civil Engineering 316.

2003 Electrical Engineering 229.

2008 Electrical Engineering 357.

2011 Industrial Engineering 355.

2020 Theoretical and Applied Mechanics 493.

2022 Civil Engineering 262.

2025 Civil Engineering 297.

2029 Music 320.

2030 Theoretical and Applied Mechanics 235.

2033 Mechanical Engineering 343.

2037 Political Science 291.

2042 Electrical Engineering 497.

2046 Theoretical and Applied Mechanics 424.

2049 Industrial Engineering 458.

2056 Urban Planning 376.

2060 General Engineering 221.

2067 Mechanical Engineering 448.

2070 Civil Engineering 374.

2071 Civil Engineering 497.

2073 Mechanical Engineering 448.

2074 Mathematics 490.

2076 Educational Psychology 497.

2080 Civil Engineering 201.

2081 Agricultural Engineering 221.

2082 Computer Science 1800.

2083 Civil Engineering 470.

2090 Accountancy 325.

2093 Architecture 348.

2096 Economics 295.

2098 Finance 294.

2106 Agronomy 440.

10. IBM 7094/1401 SERVICE, USE, AND DEVELOPMENT

Supported in part by the National Science Foundation under Grant No. NSF-GP-700.)

10.1 Log Summaries

Table I - IBM 1401-II

Summary of Use

October, 1968

Scheduled Engineering		4:15
Unscheduled Engineering		24:47
Maintenance		5:47
7094 Preparation		539:29
List/Reproduce		16:21
Code Check		:45
Tape Dump		:47
1604 Preparation		2:21
Idle		58:36
	Total	653:08

Table II - IBM 1401-II

Summary of Machine Errors

October, 1968

1402 Card Reader Punch		6
1403 Printer		1
729V Tape Drives		1
	Total	8

Table I - IBM 7094

Summary of Use

October, 1968

Scheduled Engineering			24:26
Unscheduled Engineering			3:54
Maintenance			3:11
Air Conditioning			22:55
Idle			110:06
Miscellaneous (Operator training, tape rewind, system tape mounting, rerun of failing problems)			109:41
Total Use			
Training and Education		7:25	
University Administrative Overhead Use		:41	
System Modification and Improvements		12:37	
System Accounting		8:01	
System Updating		:11	
Customer Use			
In System	303:05		
Special Short Shots	3:00		
Customer Use Total		306:05	
	Total Use		335:00
	Total Time		609:13

Table II - IBM 7094

Summary of Errors

October, 1968

Disk File		1
Console		1
Tape Drive		1
7607 Data Channel		1
	Total	4

DCSSYS[3]	0	446	446	0	10	10		20 38 23	20 38 23
SSUAD[4]	0	26	26	0	1	1		41 02	41 02
XDCS[5]	0	16	16	0	1	1		10 33	10 33
XSSS[6]	0	228	228	0	1	1		2 59 38	2 59.38
SUBTOTAL	0	716	716	0	13	13		24 29 36	24 29 36
TOTALS	940	8462	9402	20	274	294	7 24 39	327 33 18	334 57 57

1 See list of departmental codes following
2 Training and Education
3 System Improvement and Modifications
4 University Administrative Overhead Use
5 System Updating
6 Special Short Shots

Table I - IBM 1401-II

Summary of Use

November, 1968

Scheduled Engineering		4:43
Unscheduled Engineering		23:20
Maintenance		3:46
7094 Preparation		499:14
List/Reproduce		17:36
Code Check		:06
Tape Dump		4:13
1604 Preparation		2:40
Air Conditioning		2:15
Idle		67:55
	Total	625:48

Table II - IBM 1401-II

Summary of Machine Errors

November, 1968

1401 Main Frame		1
1402 Card Reader Punch		6
1403 Printer		4
	Total	11

Table I - IBM 7094

Summary of Use

November, 1968

Scheduled Engineering			21:59
Unscheduled Engineering			1:10
Maintenance			1:30
Idle			21:41
Miscellaneous (Operator training, tape rewind, system tape mounting, rerun of failing problems)			70:03
Total Use			
Training and Education		23:27	
University Administrative Overhead Use		:46	
System Improvement and Modifications		9:00	
System Accounting		8:43	
System Updating		:10	
Customer Use			
In System	248:01		
Special Short Shots	:05		
Customer Use Total		248:06	
		Total Use	290:12
		Total Time	406:35

Table II - IBM 7094

Summary of Errors

November, 1968

Disk		1
	Total	1

IBM 7094 TABLE III NOVEMBER, 1968

DEPT[1]	NUMBER OF RUNS			NUMBER OF SPECS			IBM 7094 USAGE IN HOURS-MINUTES-SECONDS			
	TANDE[2]	RES	TOTAL	TANDE[2]	RES	TOTAL	TANDE[2]	RESEARCH	REFUND	TOTAL
AAE	0	42	42	0	2	2		1 21 32		1 21 32
ADV	0	1	1	0	1	1		32		32
AGE	0	43	43	0	2	2		24 06		24 06
AGEC	195	46	241	3	6	9	43 33	1 10 28		1 54 01
AGREXT	0	2	2	0	1	1		18 21		18 21
AGRON	0	6	6	0	3	3		4 47		4 47
ANS	0	64	64	0	5	5		1 16 32		1 16 32
ARCH	64	0	64	1	0	1	10 04			10 04
ASTR	0	54	54	0	4	4		1 26 30		1 26 30
ASTRML	0	1	1	0	1	1		17 52		17 52
BINRES	0	33	33	0	2	2		1 42 07		1 42 07
CCSCS	0	1	1	0	1	1		54		54
CE	124	99	223	2	14	16	18 46	2 24 38		2 43 24
CHE	0	547	547	0	26	26		13 32 20		13 32 20
CIRCE	0	4	4	0	1	1		12 46		12 46
COMM	0	10	10	0	1	1		3 57		3 57
CRC	0	36	36	0	3	3		50 45		50 45
CS	23	1	24	1	1	2	2 45	03		2 48
ECON	0	16	16	0	2	2		10 58		10 58
ED	0	109	109	0	11	11		1 47 26		1 47 26
EDPSY	7	15	22	2	1	3	2 08	1 19 26		1 21 34
EE	0	96	96	0	9	9		1 37 50		1 37 50
ENGCST	0	5	5	0	1	1		8 52		8 52
ENTOM	0	52	52	0	2	2		33 10		33 10
FIN	0	5	5	0	2	2		1 48		1 48
FOR	0	3	3	0	1	1		43		43
FT	0	9	9	0	1	1		5 31		5 31
GENE	23	26	49	1	3	4	3 50	22 25		26 15
GECL	51	23	74	1	1	2	25 01	20 24		45 25
GSBA	0	1	1	0	1	1		3 54		3 54
HEC	0	7	7	0	2	2		10 00		10 00
HED	0	6	6	0	1	1		7 30		7 30
HORT	0	2	2	0	1	1		39		39
ICR	0	192	192	0	2	2		7 53 30		7 53 30
IEC	0	40	40	0	1	1		2 05 02		2 05 02

IBM 7C94 TABLE III NOVEMBER, 1968 -CONT

DEPT[1]	NUMBER OF RUNS TANDE[2]	RES	TOTAL	NUMBER OF SPECS TANDE[2]	RES	TOTAL	IBM 7094 USAGE IN HOURS-MINUTES-SECONDS TANDE[2]	RESEARCH	REFUND	TOTAL
IGPA	0	10	10	0	1	1		1 10 01		1 10 01
ILR	0	65	65	0	3	3		57 03		57 03
INADM	0	13	13	0	1	1		11 38		11 38
IREC	0	19	19	0	3	3		12 28		12 28
LAW	0	1	1	0	1	1		14 09		14 09
LIBS	0	1	1	0	1	1		14		14
MATH	0	18	18	0	1	1		1 36 32		1 36 32
MATRL	0	423	423	0	14	14		12 00 29		12 00 29
MCBIO	0	10	10	0	3	3		4 47		4 47
ME	3	558	561	1	25	26	1 08	23 30 25		23 31 33
MMPE	0	151	151	0	1	1		2 33 18		2 33 18
MUSIC	2	4	6	1	1	2	08	8 02		8 10
NHS	0	7	7	0	2	2		8 31		8 31
NUCE	0	140	140	0	10	10		2 30 38		2 30 38
CIR	0	278	278	0	1	1		9 30 00		9 30 00
CRME	0	7	7	0	1	1		1 09 36		1 09 36
PEM	0	11	11	0	2	2		16 58		16 58
PHYB	0	100	100	0	3	3		1 18 53		1 18 53
PHYCS	22	147	169	1	5	6	3 32	3 31 49		3 35 21
PHYX	0	1191	1191	0	10	10		105 53 38		105 53 38
POLS	128	30	158	3	3	6	17 21 17	1 47 22		19 08 39
PSYCH	230	895	1125	4	18	22	3 44 16	24 44 08		28 28 24
REC	0	8	8	0	1	1		2 56		2 56
SCCNS	0	41	41	0	1	1		15 03		15 03
SGS	0	21	21	0	1	1		12 28		12 28
SOC	76	62	138	1	5	6	10 22	4 04 39		4 15 01
SOCW	13	18	31	1	1	2	5 45	55 33		1 01 18
SPED	7	4	11	1	1	2	13 51	54		14 45
SRL	0	54	54	0	2	2		2 45 53		2 45 53
SWS	0	227	227	0	16	16		3 58 15		3 58 15
TAM	0	24	24	0	2	2		12 17		12 17
VPP	0	10	10	0	1	1		3 21		3 21
VTED	0	3	3	0	1	1		21		21
SUBTOTAL	968	6148	7116	24	258	282	23 26 26	247 59 37		271 26 03

IBM 7094 TABLE III NOVEMBER, 1968 -CONT

DEPT[1]	NUMBER OF RUNS			NUMBER OF SPECS			IBM 7094 USAGE IN HOURS-MINUTES-SECONDS			
	TANDE[2]	RES	TOTAL	TANDE[2]	RES	TOTAL	TANDE[2]	RESEARCH	REFUND	TOTAL
DCSSYS[3]	0	302	302	0	6	6		17 43 08		17 43 08
SSLAD[4]	0	30	30	0	1	1		45 54		45 54
XDCS[5]	0	20	20	0	1	1		10 22		10 22
XSSS[6]	0	8	8	0	1	1		4 40		4 40
SUBTOTAL	0	360	360	0	9	9		18 44 04		18 44 04
TOTALS	968	6508	7476	24	267	291	23 26 26	266 43 41		290 10 07

1 See list of eepartmental codes following
2 Training and Education
3 System Improvement and Modifications
4 University Administrative Overhead Use
5 System Updating
6 Special Short Shots

Table I - IBM 1401-II

Summary of Use

December, 1968

Scheduled Engineering	12:21
Unscheduled Engineering	13:15
Maintenance	2:33
7094 Preparation	509:34
List/Reproduce	9:55
Code Check	:50
Tape Dump	2:13
1604 Preparation	:10
Idle	45:14
Total	596:05

Table II - IBM 1401-II

Summary of Machine Errors

December, 1968

1402 Card Reader Punch	2
1403 Printer	1
729V Tape Drives	2
Total	5

Table I - IBM 7094

Summary of Use

December, 1968

Scheduled Engineering		30:11
Unscheduled Engineering		17:40
Air Conditioning		6:40
Maintenance		2:28
Idle		26:41
Miscellaneous (Operator training, tape rewind, system tape mounting, rerun of failing problems)		51:18
Total Use		
Training and Education	20:06	
University Administrative Overhead Use	10:25	
System Improvement and Modications	9:24	
System Accounting	8:27	
System Updating	:32	
Customer Use Total	215:59	
	Total Use	264:53
	Total Time	399:51

Table II - IBM 7094

Summary of Errors

December, 1968

7631 Disk File Control		1
Tape Drive		1
	Total	2

IBM 7094 TABLE III DECEMBER, 1968

DEPT[1]	NUMBER OF RUNS			NUMBER OF SPECS			IBM 7094 USAGE IN HOURS-MINUTES-SECONDS			
	TANDE[2]	RES	TOTAL	TANDE[2]	RES	TOTAL	TANDE[2]	RESEARCH	REFUND	TOTAL
AAE	0	17	17	0	1	1		36 32		36 32
ACCY	0	18	18	0	1	1		17 13		17 13
ADV	0	6	6	0	1	1		19 30		19 30
AGE	0	1	1	0	1	1		14		14
AGEC	349	127	476	3	6	9	1 38 01	2 46 03		4 24 04
AGREXT	0	4	4	0	1	1		2 24		2 24
AGRON	0	11	11	0	2	2		5 31		5 31
ANS	0	42	42	0	4	4		39 34		39 34
ARCH	30	0	30	1	0	1	5 27			5 27
ASTR	0	78	78	0	4	4		1 05 45		1 05 45
ASTRMI	0	5	5	0	1	1		19 48		19 48
AVI	0	2	2	0	1	1		46		46
BINRES	0	23	23	0	1	1		52 33		52 33
CCSCS	0	3	3	0	1	1		4 08		4 08
CE	133	115	248	3	11	14	21 27	5 17 36		5 39 03
CHE	0	441	441	0	24	24		14 24 09		14 24 09
CIRCE	0	5	5	0	1	1		23 31		23 31
COMM	0	21	21	0	1	1		28 51		28 51
CRC	0	18	18	0	3	3		25 25		25 25
DCS	0	4	4	0	2	2		7 29		7 29
DGS	0	16	16	0	1	1		30 10		30 10
DS	5	5	10	1	2	3	57	1 36		2 33
ECON	3	5	8	1	2	3	8 52	9 03		17 55
ED	0	75	75	0	9	9		1 06 39		1 06 39
EDPSY	229	6	235	3	1	4	1 31 43	6 14		1 37 57
EE	0	63	63	0	6	6		39 27		39 27
ENGCST	0	1	1	0	1	1		18		18
ENTOM	0	4	4	0	2	2		2 55		2 55
FIN	0	9	9	0	1	1		7 33		7 33
FOR	0	9	9	0	1	1		8 06		8 06
FT	0	7	7	0	1	1		2 24		2 24
GENE	37	39	76	1	3	4	24 07	26 19		50 26
GEOG	0	59	59	0	2	2		41 23		41 23
GEOL	34	26	60	1	1	2	33 18	1 03 54		1 37 12
GSBA	0	1	1	0	1	1		5 49		5 49

IBM 7C94 TABLE III DECEMBER, 1968 -CONT

DEPT[1]	NUMBER OF RUNS TANDE[2]	RES	TOTAL	NUMBER OF SPECS TANDE[2]	RES	TOTAL	IBM 7094 USAGE IN HOURS-MINUTES-SECONDS TANDE[2]	RESEARCH	REFUND	TOTAL
HEC	0	8	8	0	2	2		37 00		37 00
HED	0	25	25	0	1	1		23 24		23 24
HONORS	0	1	1	0	1	1		14		14
ICR	0	197	197	0	2	2		6 37 11		6 37 11
IED	0	39	39	0	1	1		6 04 51		6 04 51
IGPA	0	1	1	0	1	1		10		10
ILR	0	41	41	0	3	3		32 19		32 19
INADM	0	12	12	0	1	1		15 25		15 25
IREC	0	23	23	0	3	3		1 28 07		1 28 07
LIBS	0	3	3	0	1	1		6 18		6 18
MATH	0	4	4	0	1	1		11 45		11 45
MATRL	0	242	242	0	15	15		8 13 49		8 13 49
MCBIO	0	5	5	0	2	2		2 20		2 20
ME	56	273	329	1	8	9	22 30	10 17 39		10 40 09
MMPE	0	68	68	0	1	1		1 48 46		1 48 46
MUSIC	7	4	11	1	1	2	4 42	5 20		10 02
NHS	0	63	63	0	3	3		30 56		30 56
NUCE	0	145	145	0	7	7		2 00 22		2 00 22
OIR	0	161	161	0	1	1		4 34 37		4 34 37
CRME	0	3	3	0	1	1		13 08		13 08
PEM	0	24	24	0	5	5		33 09		33 09
PHYB	0	51	51	0	3	3		2 08 41		2 08 41
PHYCS	0	131	131	0	4	4		5 29 36		5 29 36
PHYX	0	1082	1082	0	10	10		81 54 25		81 54 25
POLS	99	26	125	4	2	6	8 14 08	1 55 33		10 09 41
PSYCH	319	906	1225	4	21	25	6 09 16	30 38 03		36 47 19
REC	0	28	28	0	2	2		1 13 04		1 13 04
SCONS	0	20	20	0	1	1		6 03		6 03
SGS	0	21	21	0	2	2		49 36		49 36
SOC	37	101	138	1	9	10	13 51	3 33 12		3 47 03
SOCW	1	10	11	1	1	2	2 06	59 06		1 01 12
SPCH	0	2	2	0	1	1		14		14
SPED	3	24	27	1	2	3	15 43	20 13		35 56
SRL	0	26	26	0	2	2		1 40 37		1 40 37
SWS	0	264	264	0	15	15		6 11 42		6 11 42
TAM	0	41	41	0	3	3		21 05		21 05

IBM 7094 TABLE III DECEMBER, 1968 -CONT

DEPT[1]	NUMBER OF RUNS TANDE[2]	RES	TOTAL	NUMBER OF SPECS TANDE[2]	RES	TOTAL	IBM 7094 USAGE IN HOURS-MINUTES-SECONDS TANDE[2]	RESEARCH	REFUND	TOTAL
VPP	0	17	17	0	1	1		7 30		7 30
VTED	0	3	3	0	1	1		2 13		2 13
ZOOL	0	33	33	0	2	2		21 31		21 31
SUBTOTAL	1342	5394	6736	27	244	271	20 06 08	215 58 06		236 04 14
DCSSYS[3]	0	461	461	0	7	7		17 50 29		17 50 29
SSUAD[4]	0		84		2	2		10 24 46		10 24 46
XDCS[5]	0	28	28	0	1	1		31 37		31 37
XSSS[6]	0	2	2	0	1	1				
SUBTOTAL	0	575	575	0	11	11		28 46 52		28 46 52
TOTALS	1342	5969	7311	27	255	282	20 06 08	244 44 58		264 51 06

1 See list of departmental codes following
2 Training and Education
3 System Improvement and Modifications
4 University Administrative Overhead Use
5 System Updating
6 Special Short Shots

7094 TABLE III 4TH QT 1968

DEPT[1]	NUMBER OF RUNS T AND E[2]	NUMBER OF RUNS RES	NUMBER OF RUNS TOTAL	NUMBER OF SPECS T AND E[2]	NUMBER OF SPECS RES	NUMBER OF SPECS TOTAL	7094 USAGE IN HOURS-MINUTES T AND E[2]	7094 USAGE IN HOURS-MINUTES RES	7094 USAGE IN HOURS-MINUTES TOTAL
AAE	0	106	106	0	5	5	0.0	3 31.2	3 31.2
ACCY	0	48	48	0	3	3	0.0	24.1	24.1
ADV	0	14	14	0	5	5	0.0	23.0	23.0
AGE	0	170	170	0	7	7	0.0	1 49.6	1 49.6
AGEC	605	214	819	9	16	25	3 0.4	4 47.8	7 48.2
AGREX	0	10	10	0	3	3	0.0	40.0	40.0
AGRON	0	19	19	0	6	6	0.0	11.0	11.0
ANS	0	168	168	0	13	13	0.0	3 4.5	3 4.5
ARCH	94	0	94	2	0	2	15.5	0.0	15.5
ASTR	0	240	240	0	13	13	0.0	4 26.5	4 26.5
ASTRM	0	8	8	0	3	3	0.0	46.7	46.7
AVI	0	2	2	0	1	1	0.0	0.7	0.7
BINRE	0	85	85	0	4	4	0.0	2 57.8	2 57.8
CCSCS	0	6	6	0	3	3	0.0	6.2	6.2
CE	583	383	966	7	41	48	1 49.3	11 45.2	13 34.4
CHE	0	1868	1868	0	75	75	0.0	56 9.5	56 9.5
CIRCE	0	13	13	0	3	3	0.0	38.0	38.0
COMM	0	33	33	0	3	3	0.0	33.7	33.7
CRC	0	73	73	0	7	7	0.0	1 33.5	1 33.5
DCS	0	5	5	0	3	3	0.0	1.6	7.5
DGS	0	16	16	0	1	1	0.0	30.1	30.1
DS	29	8	37	3	5	8	3.8	2.0	5.8
ECON	21	51	72	2	7	9	15.1	40.0	55.1
ED	0	261	261	0	30	30	0.0	4 20.8	4 20.8
EDPSY	251	21	272	7	2	9	1 52.9	1 25.6	3 18.6
EE	0	291	291	0	22	22	0.0	5 52.7	5 52.7
ENGCS	0	6	6	0	2	2	0.0	9.1	9.1
ENTOM	0	113	113	0	7	7	0.0	1 25.3	1 25.3
FIN	0	44	44	0	5	5	0.0	58.1	58.1
FOR	0	12	12	0	2	2	0.0	8.8	8.8
FT	0	28	28	0	4	4	0.0	10.9	10.9
GENE	335	105	440	3	9	12	1 34.4	1 13.7	2 48.2
GEOG	0	59	59	0	2	2	0.0	41.3	41.3
GEOL	104	104	208	3	4	7	1 5.7	3 1.6	4 7.3

7094 TABLE LII 4TH QT 1968- CONT

DEPT[1]	NUMBER OF RUNS T AND E[2]	RES	TOTAL	NUMBER OF SPECS T AND E[2]	RES	TOTAL	7094 USAGE IN HOURS-MINUTES T AND E[2]	RES	TOTAL
GSBA	C	4	4	0	3	3	0.0	11.6	11.6
HEC	C	15	15	0	4	4	0.C	47.0	47.C
HED	C	33	33	0	3	3	0.0	31.2	31.2
HLTHS	C	2	2	0	1	1	0.0	20.6	20.6
HONOR	C	2	2	0	2	2	0.0	5.6	5.6
HORT	C	4	4	0	2	2	0.C	2.2	2.2
ICR	C	523	523	0	6	6	0.0	18 47.2	18 47.2
IED	C	165	165	0	3	3	0.0	10 43.4	1C 43.4
IGPA	C	53	53	0	3	3	0.0	3 34.0	3 34.0
ILR	C	113	113	0	7	7	0.0	1 53.6	1 53.6
INADM	C	28	28	0	3	3	0.0	28.1	28.1
IREC	C	56	56	0	9	9	0.0	3 3.1	3 3.1
LAW	C	1	1	0	1	1	0.0	14.1	14.1
LIBS	C	7	7	0	3	3	0.0	16.8	16.8
MATH	C	27	27	0	3	3	0.0	2 8.1	2 8.1
MATRL	C	1484	1484	0	46	46	0.0	45 58.0	45 58.0
MCBIO	C	62	62	0	8	8	0.0	28.7	28.7
ME	85	1696	1781	3	57	60	37.8	61 8.2	61 46.0
MMPE	C	405	405	0	3	3	0.0	7 5.2	7 5.2
MUSIC	9	14	23	2	3	5	4.9	34.0	38.9
NHS	C	76	76	0	6	6	0.0	50.9	50.9
NUCE	C	421	421	0	23	23	0.0	7 25.1	7 25.1
OIR	C	809	809	0	3	3	0.0	28 23.9	28 23.9
ORME	C	13	13	0	3	3	0.0	2 8.6	2 8.6
PEM	C	54	54	0	11	11	0.0	1 26.5	1 26.5
PHYB	C	216	216	0	9	9	0.C	4 8.9	4 8.9
PHYCS	25	539	564	2	15	17	3.9	13 7.3	13 11.3
PHYX	C	3605	3605	0	30	30	0.0	310 58.2	310 58.2
POLS	239	1CC	339	9	8	17	26 15.0	10 4.2	36 19.3
PROVS	C	6	6	0	1	1	0.0	12.5	12.5
PSYCH	715	2624	3339	11	63	74	12 53.1	73 19.5	86 12.7
REC	C	42	42	0	4	4	0.0	1 18.4	1 18.4
SCONS	C	99	99	0	3	3	0.0	31.8	31.8
SGS	C	93	93	0	5	5	0.0	2 21.4	2 21.4
SOC	124	207	331	3	18	21	25.2	9 53.5	10 18.8
SOCW	21	37	58	3	3	6	10.5	2 7.4	2 18.0

7094 TABLE III 4TH QT 1969- CONT

DEPT[1]	NUMBER OF RUNS			NUMBER OF SPECS			7094 USAGE IN HOURS-MINUTES		
	T AND E[2]	RES	TOTAL	T AND E[2]	RES	TOTAL	T AND E[2]	RES	TOTAL
SPCH	0	2	2	0	1	1	0.0	0.2	0.2
SPED	10	37	47	2	4	6	29.5	32.6	1 2.2
SRL	0	154	154	0	6	6	0.0	9 25.1	9 25.1
SWS	0	709	709	0	46	46	0.0	14 28.7	14 28.7
TAM	0	77	77	0	7	7	0.0	37.8	37.8
VPP	0	46	46	0	3	3	0.0	15.5	15.5
VTED	0	11	11	0	3	3	0.0	5.6	5.6
ZOOL	0	33	33	0	2	2	0.0	21.5	21.5
Subtotal	3250	19288	22538	71	763	834	50 57.7	767 6.8	818 4.6
DCSSYS[3]	0	1209	1209	0	23	23	0.0	56 11.9	56 11.9
SSUAD[4]	0	140	140	0	4	4	0.0	11 51.7	11 51.7
XDCS[5]	0	64	64	0	3	3	0.0	52.5	52.5
XSSS[6]	0	238	238	0	3	3	0.0	3 4.3	3 4.3
Total	3250	20939	24189	71	796	867	50 57.7	839 7.2	890 5.0

1 See list of departmental codes following
2 Training and Education
3 System Improvement and Modifications
4 University Administrative Overhead Use
5 System Updating
6 Special Short Shots

10.2 Research Problem Specifications

During the fourth quarter of 1968, 49 problem specifications were submitted to the Department for computation on the 7094. The following brief descriptions of these problems have been prepared for inclusion in this report by those submitting them. T indicates a calculation associated with a thesis.

3290-8ø002 T Political Science. Illinois Criminal Courts. Analyze the similarities and differences in responses to, and adaptations to, Miranda by several types of criminal court systems in Illinois. (Neubauer)

3291-8ø003 Chemical and Metallurgical Engineering, University of Michigan. Thermodynamic Properties of Refrigerants. Compute tables of thermodynamic properties of "Freon-502". (Philip Goldblatt)

3292-8ø005 T Physical Education for Men and Graduate PE. Relationship Between Two Groups or Types of Measures. The relationship between two groups or types of measures, therefore, three different analyses: (1) a correlation matrix to determine the individual relationships between items; (2) canonical correlation; and (3) factor analysis to determine the orthogonal factors from the test variables. (Denise Allard)

3293-8ø006 T Political Science. Labor Voting Behavior. Secondary analysis of the SRC data (especially 1964) to delineate the labor voting behavior. (Ra)

3294-8ø008 Psychology. Organizational Incentive Plan. It is necessary to perform a factor analysis of employee performance data to determine which variables explain most of the performance variance. Cutting scores will be established to coincide with company standards of performance. Changes in performance and employee attitudes will be monitored and analyzed. (Andres Inn)

LIST OF DEPARTMENT CODES

If your department or office does not appear on this list, please write its full name in the department field on the Problem Specification Form even though it will require more than 6 characters.

ACCY	Accountancy
ADMREC	Admissions and Records
ADV	Advertising
AAE	Aeronautical and Astronautical Eng.
AGEC	Agricultural Economics
AGE	Agricultural Engineering
AGREXT	Agricultural Extension
AGRON	Agronomy
ANS	Animal Science
ANTH	Anthropology
ARCH	Architecture
ASTR	Astronomy
BIOPH	Biophysics
BOT	Botany
BCMPL	Bureau of Community Planning
BECBSR	Bureau of Economic Business Research
BEDRES	Bureau of Educational Research
BINRES	Bureau of Institutional Research
BED	Business Education
CZR	Center for Zoonoses Research
CERE	Ceramic Engineering
CPUBS	Champaign Public Schools
CHE	Chemistry and Chemical Engineering
CRC	Children's Research Center
CP	City Planning
CE	Civil Engineering
COMM	Communications
CURLAB	Curriculum Laboratory
DS	Dairy Science
DCS	Department of Computer Science
DGS	Division of General Studies
DUE	Division of University Extension
DOW	Division of Waterways
ECON	Economics
ED	Education
EDPSY	Educational Psychology
EDADM	Educational Admin. and Supervision
EDTEST	Educational Testing
EE	Electrical Engineering
ENGADM	Engineering Administration
ENGCST	Engineering College and Station
ENGH	Engineering Honors Program
ENGLSH	English
ENTOM	Entomology
EDC	Extension Division Counseling
FIN	Finance
FT	Food Science
FOR	Forestry
GENE	General Engineering
GEOG	Geography
GEOL	Geology
GER	German
GSBA	Graduate School of Business Admin.
HED	Health Education
HLTHSV	Health Service
HEC	Home Economics
HONORS	Honors Program
HORT	Horticulture
ILLDMH	Illinois Dept. of Mental Health
INADM	Industrial Administration
IE	Industrial Engineering
IREC	Institute for Research on Exc. Children
ICR	Institute of Communications Research
IGPA	Institute of Govt. and Public Affairs
ILR	Institute of Labor and Ind. Relations
LIBS	Library Science
LING	Linguistics
MKTG	Marketing
MATRL	Materials Research Laboratory
MATH	Mathematics
ME	Mechanical Engineering
MCBIO	Microbiology
MMPE	Mining, Metallurgy, and Petroleum Eng.
MUSIC	Music
NHS	Natural History Survey
NUCE	Nuclear Engineering
OAC	Office of Agricultural Communication
DNW	Office of the Dean of Women
OIR	Office of Instructional Resources
PEM	PE for Men and Graduate PE
PEW	PE for Women
PHYPLA	Physical Plant
PHYCS	Physics
PHYB	Physics Betatron Laboratory
PHYX	Physics Project X
PHYSL	Physiology and Biophysics
PLPA	Plant Pathology
POLS	Political Science
PROVST	Provost's Office
PSYCH	Psychology
REC	Recreation
SHCBRC	Small Homes Council
SOCW	Social Work
SOC	Sociology
SCONS	Soil Conservation Service
SPED	Special Education
PSDEC	Special Education, Decatur Pub. Schools
SPCH	Speech and Theatre
SGS	State Geological Survey
SWS	State Water Survey
SCS	Student Counseling Service
SRL	Survey Research Laboratory
TAM	Theoretical and Applied Mechanics
USGS	U.S. Geological Service
UNIHI	University High School
UCCTE	Urbana-Champaign Coun. on Teacher Ed.
VMS	Veterinary Medical Science
VMA	Veterinary Medicine Administration
VPH	Veterinary Pathology and Hygiene
VPP	Veterinary Physiology and Pharmacology
VTED	Vocational and Technical Education
WPGU	WPGU Radio Station
ZOOL	Zoology

Chicago Circle

CCCHE	Chemistry
CCDME	Materials Engineering
CCENEN	Energy Engineering
CCPHCS	Physics
CCSOC	Sociology
CCSCS	Student Counseling Service

Medical Center

ORME	Office of Research in Medical Ed.
OT	Occupational Therapy

Ill. State University

ISEDAD	Department of Education Administration

3295-8ø010 Sociology. Small Claims Court. This is a study of the small claims court using a sample of 500 past users of the small claims court of Champaign, Illinois. It seeks to answer questions such as: who uses the court, for what, and how did they learn of this procedure. (Kronus)

3296-8ø012 T Sociology. Chicago Policewomen Study. Study of policewomen in Chicago area - exploratory research into attitudinal and behavioral variables. (Leslie Pfister)

3297-8ø013 Education. Junior College Student Study. A sample of 200 junior college students responded to twenty pairs of bipolar adjectives for each of seven criterion words on a semantic differential instrument. The D^2 statistics and factor analytic methods will be used to analyze these responses. The factor analyses will be used to ascertain the connotative framework used by this sample of students and by subgroups of the sample. The D^2 statistic will be used to gauge any clustering of the criterion words and the differences which may occur as this same sample responds to the instrument at a future date. (William Wellner)

3298-8ø014 T Physical Education for Men and Graduate PE. The Speed Factor in the Illinois Agility Run. The purpose of this problem is to determine the amount of variance in the Illinois Agility Run that can be accounted for by the speed factor. The statistical procedure will be correlation of dash time and Illinois Agility Run time. (Paula L. Schutt)

3299-8ø015 T Education. The Academic Performance of Culturally Disadvantaged Adolescents. Teacher and student academic expectations were experimentally manipulated with a population of culturally disadvantaged adolescents. It is hypothesized that the parameters related to academic success will be a function of the experimentally induced expectation levels. The experimental design is a three-way multi-variate analysis of variance with approximately 15 dependent measures. (T. H. Anderson)

3300-8ø016 Graduate School of Business Administration. Purchasing Behavior Predicted by Need for Achievement. The relationship between levels of need for achievement and present and desired inventory of goods and services. (David M. Gardner)

3301-8∅018 Psychology. Multivariate Study of Form Perception. Multivariate analysis of data from a set of experiments on form perception. SSUPAC programs for factor analysis, multiple discriminant analysis, canonical correlation, and distribution analysis will be used. In addition, FORTRAN II programs for nonmetric multidimensional sealing will be employed. (Larry G. Richards)

3302-8∅019 T Education. Transfer and Inhibition in Free Recall of Verbal Material. Lists of words which either could or could not be easily organized in recall were studied to criterion by college students. Interpolated lists of the same character were presented for four trials before recall of original learning. Effects of transfer and inhibition were studied. (Graeme Watts)

3303-8∅022 Civil Engineering. STRESS. Structural analysis. (S. J. Fenves)

3304-8∅023 Economics. Secondary Wage Changes. To determine the amount and pattern of wage emulation over the business cycle between manufacturing industries. (Wallack)

3305-8∅024 Psychology. Distribution of Fears in College Students. This problem will involve frequency counts and factor analysis. The data represent reports of one of seven levels of fear with respect to 51 stimuli or situations. We will want to know the fear-level pattern for each of approximately 2500 subjects, the number of times each item was ranked at each level, and the ways in which items tended to cluster together in terms of their ability to elicit fear reports. (D. A. Bernstein)

3306-8∅025 Geography. Evaluation of Geography 101. Analysis of student reactions to Geography 101 course contant and instructional methods. Anticipate use of SSUPAX programs--frequency count, missing data correlations, principal axis factor analysis with verimax rotation. (Janice Monk)

3307-8ø027 T Psychology. Sequential Effects in Absolute Judgment. Data has been collected suggesting that accuracy of size judgments is related to the order in which the stimuli are seen. A mathematical model stemming from multiple discriminant analysis is to be applied to these data providing a useful description of the judgmental process and a measure of the degree to which sequential effects are present. (Friden)

3308-8ø028 T Educational Psychology. Cognitive Systems in Judgment of Teaching Performance. Four separate principal components analyses will be performed on minor product matrices formed from input matrices of order N x 26, two N x 210, and N x 84; N 100. Output includes characteristic roots and vectors and N x r and n x r matrices of loadings on principal components (n = 26, 210, 84). Loadings of subjects on principal components from first three analyses will be dependent variables multivariate analysis of variance and predictors in canonical correlation analysis with loadings from fourth analysis as criteria. (Rumery)

3309-8ø029 T Home Economics. Inhibition of Infant Distress. Distressed infants usually become quiet when rocked or taken in a car for a ride. This effect could be due to motion itself or some environmental-social factor related or associated with motion. The present study concerns the effect of various speeds of motion and presence or absence of mother for infants in four age groups from 3 to 25 weeks of age. While various correlated of infant responsiveness to motion will be investigated, the primary design corresponds to an age by sex by treatments, repeated measures ANOVA of difference scores (for pre- and post tests). This analysis may be supplemented by a treatment by levels AN OV or an ANCOV depending upon distribution of scores for pretest. SSUPAC is the intended program for analysis (Van Den Daele)

3310-8ø030 Institute of Labor and Industrial Relations. Determinants of Rule Performance. Analyses of the relationships between the variables associated with a member's psychological role and his role performance within an organization. Analysis requires SSUPAC. (George B. Graen)

3311-8Ø031 Law. Analysis of Automobile Insurance. Frequency tabulations using Survey Research Laboratory 1968 Omnibus Survey data on attitudes toward auto insurance. (J. O'Connell)

3312-8Ø032 Geography. SMSA Gravity Model. An attempt to measure and predict, through regression analysis and associated techniques, the existant relationships between population densities and the movement of goods among and between the S.M.S.A.'s in the United States. (Roepke)

3313-8N001 Graduate School of Business Administration. Economics of Advertising. Economics of advertising - several related regression studies. (Julian L. Simon)

3314-8N002 Home Economics. A Longitudinal Study of Vocal Development. The basic problem in this study is a longitudinal project involving the later effects of group nursery school music training. Related problems include vocal acceleration, multi-sensory performance practice, and the effect of vocal range on singing performance. (Robert B. Smith)

3315-8N003 T Physical Education for Men and Graduate PE. The Long-Term Effects of an Exercise Program on Selected Physiological and Psychological Measures in Middle-Aged Men. The purpose of this study is to determine the long-time value of exercise programs on selected physiological and psychological parameters in non-athletic, middle-aged men engaged in sedentary occupations. (Jette)

3316-8N004 Special Education. Acquisition of a Complex Assembly Task by Retardates. To determine the effects of certain learning variables on retardate performance in a sheltered workshop. (Marc W. Gold)

3317-8N005 Psychology. Follow-up Evaluation of Volunteer Project Effects. Evaluation of outcome results for various sub-groups of patients and students who participated in a volunteer state hospital project. Investigation of the relationship between patient, volunteer, attendance, and activity variables. (Julian Rappaport)

3318-8N006 Survey Research Laboratory. Non-Charge Development Runs. Development of programs for future survey to be run. (M. F. Uchida)

3319-8N007 Children's Research Center. Social Factors in Children's Play. To determine the effects of competition, failure, and knowledge of results, on aspiration level and motor performance. Basic statistics needed for analysis of multiple factorial designs, e.g., analysis of variance, chi squares, and correlations. (M. J. Ellis)

3320-8N009 T Zoology. Rehavioral Changes Due to Cross-Fostering. An analysis of variance will be run comparing normal mice with those raised by another species. (David Quadagno)

3321-8N013 Psychology. Individual Differences in Multiple Cue Learning. A search for clusters of individuals in a multiple cue learning task and the relation of these clusters to psychological measurements of these individuals. (Schenck)

3322-8N014 Agricultural Economics. Husband-Wife Response Discrepancy. Data from 300 married couples in Brasilia, Brazil, are used to compare discrepancy between husband's response and wife's response to identical questions. Response discrepancy will then be related to personal characteristics of the respondent. The study has mainly methodological objectives. (J. C. Van Es)

3323-8N015 Education. Thai Universities. The research concerns the sociology of universities and students in Thailand. Social mobility mediated by the universities is examined for several of its aspects, including: rural-urban migration, ethnic assimilation, and class mobility. The research studies the campus culture which emerges among the students and the impact it has on the attitudes of the students. Research is directed also to the investigation of organizational change occurring in the universities in response to technological and other sources of modernization existing in the larger society. (William E. Maxwell)

3324-8N016 T Education. Attitudes Toward Science. The study will use the semantic differential technique to determine how the meanings of concepts change during a course in college physical science. An attempt will also be made to determine whether various sub-groups within the main group have different semantic structural frameworks. (John O. Wernegreen)

3325-8D001 Psychology. Imitation and Comprehension of Adult Sentences by Young Children. This is a study in the area of developmental psycholinguistics. The study examines young children's, age 3 to 5, ability to imitate and comprehend sentences of various types. (Gramling)

3326-8D004 Psychology. Doctor of Psychology Program Evaluation. Factor analysis of clinical trainee competence ratings, and additional problems relating to evaluation of the Doctor of Psychology program. (Dr. D. Peterson)

3327-8D005 T Finance. Option Premiums. Transformation program on option premiums. (Silber)

3328-8D006 Sociology. Attitude Formation Research Based on WISOB. This research involves the calculation of the relative amounts of influence over the attitudes of an individual exercised by significant others, self-reflexive acts and related cognitive structures. (Woelfel)

3329-8D007 Education. Student Opinionaire Development. A study of opinions of students on various subjects is to be analyzed in an attempt to develop a Student Opinionaire form. (Thomas)

3330-8D008 Physical Education for Men and Graduate PE. Individual and Group Performance on a Muscular Power Task. Individuals will be measured on their rope pulling power. These same individuals will then pull the rope in groups of 2's, 4's, and 8's. The fatigue and strength factors will be controlled. The potential productivity of the group (the sum of what the individuals did alone) will then be compared to the actual productivity of the various groups to determine the extent of the coordination and/or motivation losses in the initial learning and later performance on this additive task. (D. Landers)

3331-8D009 Advertising. Source and Content of Communication About Advertising and That Effect on Student Attitudes About Communication. 2 x 2 factorial design which investigates the effect of the source and content of communications about advertising on student attitudes about advertising. (Wiggins)

3332-8D010 Provost's Office. Organizational Study of Departmental Performance. This project is an investigation of organizational factors of university academic departments and how these factors relate to various departmental outputs, such as quality of graduate students and publications of the faculty. (Lanier)

3333-8D012 T Special Education. The Acquisition of a Complex Asembly Task by Retardates. Use of SSUPAC programs for ANOVA to determine effectiveness of techniques to train trainable retardates to do complex assembly work. (Marc W. Gold)

3334-8D013 Speech and Theatre. Variables Related to Belief Structure. An attempt will be made to extract variables related to belief structure through statistical analysis. (R. A. Clark)

3335-8D014 T Psychology. Effects of Hypnotic Suggestion on Pain Perception. An attempt to assess the roles of the variables of hypnotic induction and analgesic suggestion in pain relief using two autonomic indices of pain perception (heart rate and finger pulse volume) and subjective report. (Evans)

3336-8D015 Aviation. Analysis of Optimized Training Program. Analysis of scores obtained from pilots' performance in airline simulators and in aircraft to determine the effectiveness of simulators in the transition of airline captains to new aircraft. (Gandelman)

3337-8D016 T Advertising. Retail Advertising and Store Image. Research designed to determine if, in fact, there is a positive relationship between images perceived through advertising (specifically newspaper advertisements) and the consumer image of the retail outlet which put the ad in the newspaper. (Tate)

3338-8D019 T Education. Cognitive and Perceptual Skills of Conduct Problem Children. Forty-seven children rated as conduct problems by their classroom teachers were given an extensive battery of perceptual and cognitive tasks to determine how they differ on the performance of these tasks from the normal population. (Loyd S. Wright)

10.3 Class Problem Specifications

During the fourth quarter of 1968, 19 problem specifications were submitted to cover all assigned problems on the 7094 in the following courses.

J955-8Ø001	Educational Psychology 390.
J956-8Ø004	Geology 479.
J957-8Ø007	Music 304.
J958-8Ø009	Psychology 332.
J959-8Ø011	General Engineering 103.
J960-8Ø017	Political Science 326.
J961-8Ø020	Special Education 449.
J962-8Ø021	Sociology 185.
J963-8Ø026	Dairy Science 205.
J964-8N008	Music 320.
J965-8N010	Political Science 326.
J966-8N011	Political Science 291.
J967-8N012	Aeronautical and Astronautical Engineering 241.
J968-8N017	Educational Psychology 497.

J969-8D002	Mechanical Engineering 293.
J970-8D003	Civil Engineering 297.
J971-8D011	Economics 295.
J972-8D017	Political Science 345.
J973-8D018	Sociology 385.

11. GENERAL DEPARTMENT INFORMATION

11.1 Personnel

The number of people associated with the Department in various capacities is given in the following table:

	Full-time	Part-time	Full-time Equivalent
Faculty	23	2	24.30
Visiting Faculty	2	0	2.00
Research Associates	4	0	4.00
Graduate Research Assistants	13	110	70.46
Graduate Teaching Assistants	0	8	3.00
Professional Personnel	29	3	31.17
Administrative and Clerical	27	0	27.00
Nonacademic Personnel (Monthly)	63	3	64.75
Nonacademic Personnel (Hourly)		119	36.02
TOTAL	161	245	262.70

The Department Advisory Committee consists of Professor J. R. Pasta, Head of the Department; Professor J. N. Snyder, Associate Head of the Department; Professors K. W. Dickman, M. Faiman, L. D. Fosdick, H. G. Friedman, C. W. Gear, D. B. Gillies, D. J. Kuck, B. H. McCormick, S. Muroga, T. A. Murrell, J. Nievergelt, R. S. Northcote, J. R. Phillips, W. J. Poppelbaum, S. R. Ray, J. E. Robertson, P. E. Saylor, and D. L. Slotnick.

11.2 Bibliography

During the fourth quarter, the following publications were issued by the Laboratory:

File Numbers

(1) Dill, C., Ellis, C. A., Gear, C. W. and Ratliff, K., "The Automatic Integration Package for Ordinary Differenital Equations," File No. 779, November, 1968

(2) Flowerdew, S., "Automatic Karyotyping: A Statement of the Problem and Summary of Image Processing Procedures," File No. 780, November 6, 1968

(3) Gear, C. W., "Generalized Simulation and Modeling," File No. 785, December 30, 1968

(4) Katoh, L., "Task Directive Language for the Illiac III Computer," File No. 782, December, 1968

(5) Krabbe, S. P., "Specifications for Termi-Point Wiring of Section 6 (Taxicrinic Processing) of the Illiac III," File No. 550-118, October 29, 1968

(6) Krabbe, S. P., "Specifications for a 9" False Floor for DCL 223 that Overlaps the Existing False Floor in DCL 280," File No. 550-120, November 5, 1968

(7) Krabbe, S. P., "Specifications for Wire Wrap of Section 13 (Exchange Net)," File No. 550-119, November 6, 1968

(8) Krabbe, S. P., "Specifications for Wire Wrap of Section 14," File No. 550-121, November 5, 1968

(9) McCarthy, T. E., "Solving a Set of Coupled Systems of Ordinary Differential Equation on Illiac IV," File No. 778, November 5, 1968

(10) Rudsinski, L. E., "A Tranquil Version of an Idealized Particle-In-Cell Code for a Plasma," File No. 781, November 25, 1968

(11) Rudsinski, L. E., "Tranquil Code for the Modified Successive Overrelaxation (MSOR) Method for Solving Laplaces Difference Equation," File No. 784, December 24, 1968

(12) Wallman, L. H., "Camera Specifications," File No. 550-122, December 5, 1968

(13) Winje, G. L., "'Massless' Particle Trace and the Numerical Solution of the Eulerian Hydrodynamic Equations on Illiac IV," File No. 777, October 22, 1968

Report Numbers

(1) Flowerdew, S. J., "LEFT: A Language for Editing and Formating Text," Report No. 291, October 30, 1968

(2) "Illiac IV Progress Report - July -- September 1968," Report No. 294

(3) Koo, P. L. and Atkins, D. E., "Arithmetic Unit of Illiac III Simulation and Logical Design -- Part II," Report No. 290, October 28, 1968

(4) Muroga, S. and Ibaraki, T., "Logic Design of an Optimum Network by Interlinear Programming Part II," Report No. 289, October, 1968

Thesis

(1) Liu, T., "A Code for Zero-One Integer Linear Programming by Implicit Enumeration," (A Programming Manual for ILLIP), Report No. 302, December 15, 1968

11.3 Colloquia

"Logical Design with One NOT - Element, or How to Say 'NO' Once and Really Mean It," by David Huffman, University of California, Santa Cruz, California, October 21, 1968

"Boolean Analyzer and Its Application in Design Automation," by Antonin Svoboda, University of California, Los Angeles, California, November 4, 1968

"Do Most Machines Have a Series-Parallel Decomposition?," by David Muller, University of Illinois, Urbana, Illinois, November 18, 1968

"Galerkin Methods for Parabolic Equations," by Jim Douglas, Jr., University of Chicago, Chicago, Illinois, November 25, 1968

11.4 Shops' Production

Fabrication Facility Annual Report by Contracts is as follows:

	1018	1469	300	360/7094	4144
WORK REQUESTS COMPLETED					
Machine Shop	88	63	50	9	9
Electronic Shop	132	109	7	0	1
Chemical Shop	41	112	31	5	1
Layout Shop	81	105	6	0	2
TOTAL	342	389	94	14	13

Fabrication Facility Printed Wiring Board Annual Report is as follows:

	1018	1469P	1834
Printed Boards Wired	856	427	37
Printed Boards Converted	536	0	0
Printed Boards Repaired	32	0	0
Transistors Wired	9207	9236	318
Diodes Wired	47697	9639	837
Integrated Circuits Wired	34	2470	69

(F. Serio)

11.5 Drafting

During the fourth quarter, a total of 344 drawings were processed by both drafting sections:

	General	Pattern Recognition
Large Drawings	62	72
Medium Drawings	27	60
Small Drawings	37	19
Layouts	17	0
Report Drawings	33	0
Changes	1	8
Miscellaneous	8	0
TOTAL	185	159

(M. Goebel and J. Otten)